Large-Scale Light Field Capture and Reconstruction

M.Sc. Yuan Gao

Dissertation
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
(Dr.-Ing.)
der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel
eingereicht im Jahr 2020

Kiel Computer Science Series (KCSS) 2020/2 dated 2020-07-15

URN:NBN urn:nbn:de:gbv:8:1-zs-00000367-a4
ISSN 2193-6781 (print version)
ISSN 2194-6639 (electronic version)

Electronic version, updates, errata available via https://www.informatik.uni-kiel.de/kcss

The author can be contacted via https://www.mip.informatik.uni-kiel.de

Published by the Department of Computer Science, Kiel University

Multimedia Information Processing Group

Please cite as:

▻ Yuan Gao. *Large-Scale Light Field Capture and Reconstruction*. Number 2020/2 in Kiel Computer Science Series. Department of Computer Science, 2020. Dissertation, Faculty of Engineering, Kiel University.

```
@book{Gao20,
  author    = {Yuan Gao},
  title     = {Large-Scale Light Field Capture and Reconstruction},
  publisher = {Department of Computer Science, Kiel University},
  year      = {2020},
  number    = {2020/2},
  doi       = {10.21941/kcss/2020/2},
  series    = {Kiel Computer Science Series},
  note      = {Dissertation, Faculty of Engineering,
               Kiel University.}
}
```

About this Series

The Kiel Computer Science Series (KCSS) covers dissertations, habilitation theses, lecture notes, textbooks, surveys, collections, handbooks, etc. written at the Department of Computer Science at Kiel University. It was initiated in 2011 to support authors in the dissemination of their work in electronic and printed form, without restricting their rights to their work. The series provides a unified appearance and aims at high-quality typography. The KCSS is an open access series; all series titles are electronically available free of charge at the department's website. In addition, authors are encouraged to make printed copies available at a reasonable price, typically with a print-on-demand service.

Please visit `http://www.informatik.uni-kiel.de/kcss` for more information, for instructions how to publish in the KCSS, and for access to all existing publications.

1. Gutachter:	Prof. Dr.-Ing. Reinhard Koch Christian-Albrechts-Universität zu Kiel Kiel
2. Gutachter:	Prof. Dr. Atanas Gotchev Tampere University Tampere

Datum der mündlichen Prüfung: 23. Juni 2020

Zusammenfassung

Diese Arbeit diskutiert Verfahren zur Konvertierung von Sparsely-Sampled Light Fields (SSLFs) zu Densely-Sampled Light Fields (DSLFs), welche Anwendung in den Bereichen 3DTV und Virtual Reality (VR) finden. Beispielhaft wird hierbei ein bewegliches 1D-Lichtfeldakquisitionssystem zur Aufnahme von SSLFs in realen Umgebungen evaluiert. Dieses System besteht aus 24 RGB-Kameras und zwei Kinect V2 Sensoren. Die damit aufgenommenen SSLF Daten können zur Rekonstruktion von DSLF genutzt werden. Zu diesem Zweck müssen drei wesentliche Probleme gelöst werden: (i) Die Schätzung einer rigiden Transformation zwischen den Koordinatensystemen einer Kinect V2 und einer RGB-Kamera; (ii) Die Registrierung von zwei Kinect V2 Sensoren, welche mit einer großen Distanz zueinander platziert sind; (iii) Die Rekonstruktion eines DSLF aus einem SSLF mit moderaten bis großen Disparitäten.

Um diese drei Probleme zu lösen, wurden folgende Verfahren entwickelt: (i) Eine neuartige Selbstkalibrierung, welche die geometrischen Beschränkungen der Szene und Kameras nutzt, um rigide Transformation von dem Koordinatensystem einer Kinect V2 zu den Koordinatensystemen der 12 nächstgelegenen RGB-Kameras zu schätzen; (ii) Ein neuartiger grob-zu-fein Ansatz zur Schätzung der rigiden Transformation zwischen den Koordinatensystemen zweier Kinect V2 Kameras anhand lokaler Farb- und Geometrieinformationen; (iii) Verschiedene neue Algorithmen zur Rekonstruktion von DSLFs aus SSLFs, welche in zwei Gruppen eingeordnet werden können. Zum einen die Synthese neuer Ansichten inspiriert durch aktuelle Video-Interpolationsmethoden und zum anderen das Inpainting in Epipolar Plane Images (EPIs), welches von Rekonstruktionsmethoden basierend auf der Shearlet Transformation (ST) inspiriert wurden.

Abstract

This thesis discusses approaches and techniques to convert Sparsely-Sampled Light Fields (SSLFs) into Densely-Sampled Light Fields (DSLFs), which can be used for visualization on 3DTV and Virtual Reality (VR) devices. Exemplarily, a movable 1D large-scale light field acquisition system for capturing SSLFs in real-world environments is evaluated. This system consists of 24 sparsely placed RGB cameras and two Kinect V2 sensors. The real-world SSLF data captured with this setup can be leveraged to reconstruct real-world DSLFs. To this end, three challenging problems require to be solved for this system: (i) how to estimate the rigid transformation from the coordinate system of a Kinect V2 to the coordinate system of an RGB camera; (ii) how to register the two Kinect V2 sensors with a large displacement; (iii) how to reconstruct a DSLF from a SSLF with moderate and large disparity ranges.

To overcome these three challenges, we propose: (i) a novel self-calibration method, which takes advantage of the geometric constraints from the scene and the cameras, for estimating the rigid transformations from the camera coordinate frame of one Kinect V2 to the camera coordinate frames of 12-nearest RGB cameras; (ii) a novel coarse-to-fine approach for recovering the rigid transformation from the coordinate system of one Kinect to the coordinate system of the other by means of local color and geometry information; (iii) several novel algorithms that can be categorized into two groups for reconstructing a DSLF from an input SSLF, including novel view synthesis methods, which are inspired by the state-of-the-art video frame interpolation algorithms, and Epipolar-Plane Image (EPI) inpainting methods, which are inspired by the Shearlet Transform (ST)-based DSLF reconstruction approaches.

Acknowledgements

First of all, I would like to thank my advisor Prof. Dr.-Ing. Reinhard Koch who offered me to pursue a Ph.D. at the Multimedia Information Processing (MIP) group. His support, trust and the freedom to investigate new light field-associated research fields were invaluable during my Ph.D.

I would like to thank Prof. Dr. Atanas Gotchev for accepting to be in my reading committee. I am also grateful to the other members of the jury Prof. Dr.-Ing. Dirk Nowotka and Prof. Dr.-Ing. Sven Tomforde.

I would like to express my heartfelt gratitude to the current members of the MIP group (Johannes Brünger, Dr.-Ing. Vasco Grossmann, Tim Michels, Luca Palmieri, Dr.-Ing. Arne Petersen, Stefan Reinhold, Lars Schmarje, Simon-Martin Schröder, Tobias Schwede, Dr.-Ing. Christoph Starke, Renate Staecker, Torge Storm and Dr.-Ing. Claudius Zelenka) and my former colleagues (Sascha Clausen, Dr.-Ing. Sandro Esquivel, Dr.-Ing. Oliver Fleischmann, Yu Tang and Dr.-Ing. Dominik Wolters). I would like to thank Tim Michels for the translation of the abstract into German.

Furthermore, I would like to thank Dr. rer. nat. Frederik Zilly and Dr.-Ing. Joachim Keinert for offering me the opportunity to carry out collaborative research at Fraunhofer IIS, Erlangen, for four months. I am sincerely thankful to Prof. Dr. Atanas Gotchev, Dr. Robert Bregovic and Dr. Suren Vagharshakyan for their support, advice and fruitful discussion during my secondment at Tampere University of Technology (TUT), Tampere, for two and a half months.

Finally, I am deeply thankful to my friends and family, particularly to my parents and grandparents, who supported me from the beginning.

The research work in this thesis was supported by the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging.

Contents

List of Figures

List of Abbreviations

3DTV	3D television
BA	Bundle Adjustment
CNN	Convolutional Neural Network
CT	Computed Tomography
DFT	Discrete Fourier Transform
DIBR	Depth-Image-Based Rendering
DORE	double overrelaxation
DRN	Disparity Refinement Network
DRST	Deep Residual Shearlet Transform
DSLF	Densely-Sampled Light Field
DSLR	Digital Single-Lens Reflex camera
EPI	Epipolar-Plane Image
FAST	Flow-Assisted Shearlet Transform
FoV	Field of View
FTV	free viewpoint television
HDCA	High Density Camera Array
IBR	Image-Based Rendering
ICP	Iterative Closest Point
IEST	Interpolation-Enhanced Shearlet Transform
KNN	*k*-Nearest-Neighbors
LSTM	Long Short-Term Memory
MAST	Mask-Accelerated Shearlet Transform

MIP	Multimedia Information Processing
MPI	Multiplane image
PIASC	Parallax-Interpolation Adaptive Separable Convolution
P*n*P	Perspective-*n*-Point
RANSAC	RANdom SAmple Consensus
SepConv	Separable Convolution
SIFT	Scale-Invariant Feature Transform
SSLF	Sparsely-Sampled Light Field
ST	Shearlet Transform
SURF	Speeded Up Robust Features
SVD	Singular Value Decomposition
TDF	Truncated Distance Function
ToF	Time-of-Flight
TSDF	Truncated Signed Distance Function
VR	Virtual Reality
VSN	View Synthesis Network

Symbols and Notations

$\mathbb{R}$	set of all real numbers
$\mathbb{Z}$	set of all integers
$\mathbb{R}_+$	set of all positive real numbers
$\mathbb{N}^*$	set of all natural numbers (excluding 0)
$\mathbb{R}^n$	Euclidean space of dimension n
$\mathbb{P}^n$	projective space of dimension n
$\mathbb{SO}(n)$	special orthogonal group of order n
$\mathbb{SE}(n)$	special Euclidean group in dimension n
$\boldsymbol{P} = \begin{pmatrix} P_x, & P_y, & P_z, & 1 \end{pmatrix}^{\mathrm{T}}$	3D point with its homogeneous coordinates
$\tilde{\boldsymbol{P}} = \begin{pmatrix} P_x, & P_y, & P_z \end{pmatrix}^{\mathrm{T}}$	3D point with its Cartesian coordinates
$\mathbf{R} \in \mathbb{SO}(3)$	3D rotation matrix
$\boldsymbol{t} \in \mathbb{R}^3$	3D translation vector
$\mathbf{T} \in \mathbb{SE}(3)$	rigid (Euclidean) transformation
$\mathbf{K}$	camera intrinsic matrix
$\mathrm{proj}(\cdot)$	perspective projection function
$\mathbb{C}_1, \mathbb{C}_2, \ldots, \mathbb{C}_{24}$	RGB cameras
$\mathbb{C}_A, \mathbb{C}_B$	Kinect V2 sensors
$\kappa(\cdot, \cdot)$	camera back-projection function
$\pi(\cdot)$	camera projection function
u, v	axes of image plane
s, t	axes of camera plane
$\mathcal{S} = \{\mathcal{I}_j \vert 1 \leqslant j \leqslant n\}$	3D SSLF composed of parallax images
$\mathcal{S} = \{\varepsilon_i \vert 1 \leqslant i \leqslant l\}$	3D SSLF composed of sparsely-sampled EPIs
$\mathcal{D} = \{\tilde{\mathcal{I}}_k \vert 1 \leqslant k \leqslant \dot{n}\}$	3D DSLF composed of parallax images
$\mathcal{D} = \{\zeta_i \vert 1 \leqslant i \leqslant l\}$	3D DSLF composed of densely-sampled EPIs
$m \times l$	spatial resolution of $\mathcal{S}$ and $\mathcal{D}$
n	(i) the number of point pairs in Chapter 3 (ii) angular resolution of $\mathcal{S}$ in Chapter 4
$\dot{n}$	angular resolution of $\mathcal{D}$

$\mathcal{I}_j \in \mathbb{R}^{m \times l \times 3}$	color parallax image of $\mathcal{S}$
$\tilde{\mathcal{I}}_k \in \mathbb{R}^{m \times l \times 3}$	color parallax image of $\mathcal{D}$
$\varepsilon_i \in \mathbb{R}^{m \times n \times 3}$	color sparsely-sampled EPI of $\mathcal{S}$
$\zeta_i \in \mathbb{R}^{m \times \dot{n} \times 3}$	color densely-sampled EPI of $\mathcal{D}$
d_{min}	minimum disparity of $\mathcal{S}$
d_{max}	maximum disparity of $\mathcal{S}$
d_{range}	disparity range of $\mathcal{S}$
τ	sampling interval
ϱ	size of separable 1D vectors of SepConv
$*$	convolution operation
$\odot$	element-wise (Hadamard) product
%	modulo operation
$g(\cdot, \cdot)$	inverse warping function
$\mathcal{F}_{j \rightarrow (j+1)}$	optical flow map from $\mathcal{I}_j$ to $\mathcal{I}_{j+1}$
$\mathcal{F}^u_{j \rightarrow (j+1)}$	disparity map from $\mathcal{I}_j$ to $\mathcal{I}_{j+1}$
t	space step
$\mathcal{V}_{(j+t) \leftarrow j}$	soft visibility map from $\mathcal{I}_j$ to $\mathcal{I}_{j+t}$
φ	shearing parameter
ξ	the number of shearlet scales
η	the number of shearlet filters
$\mathcal{SH}(\cdot)$	shearlet analysis transform
$\mathcal{SH}^*(\cdot)$	shearlet synthesis transform
$\gamma \times \gamma$	spatial resolution of shearlet filters
d	the number of iterations
$\mathrm{sum}(\cdot)$	sum function
λ_i	threshold level
$T_{\lambda_i}(\cdot)$	hard-thresholding function using λ_i
$\mathcal{R}(\cdot)$	encoder-decoder network
δ	interpolation rate
$\mathcal{E}$	3D light field reconstructed from $\mathcal{S}$ using δ
$1 < d_{range} \leqslant 8$ pixels	small disparity range
$8 < d_{range} \leqslant 16$ pixels	moderate disparity range
$d_{range} > 16$ pixels	large disparity range
$\mathcal{S}^{4\mathrm{D}}$	4D SSLF
$\mathcal{D}^{4\mathrm{D}}$	4D DSLF

Chapter 1

Introduction

1.1 Motivation

Nowadays, with more and more 3D television (3DTV) [WOO+19; Smo11; TTF+11; MP04], Virtual Reality (VR) [OEE+18; Yu17] and holographic [Yam16; Bal06; ABF+06] devices having been launched into the consumer entertainment market, how to produce low-cost and high-quality contents for these devices is attracting increasing attention from both scientific and industrial communities. Light field [LH96; GGS+96] is one of the most promising techniques that can achieve this goal via capturing the light rays coming from different locations and directions in real-world scenes. Specifically, a 4D light field is an approximation of the plenoptic function parameterized by two parallel planes, i. e., camera plane and image plane. Many visualization applications on 3DTV, VR and holographic devices are aimed at giving people an authentic experience in an immersive environment by rendering realistic contents sampled from large real-world scenes. Designing and constructing a large-scale light field acquisition system is therefore important for these applications. To this end, the most straightforward way would be to build a 2D camera array with a big size having a lot of RGB cameras that are densely and uniformly distributed on the camera plane, so that the data of the static or dynamic 4D Densely-Sampled Light Fields (DSLFs) of the large real-world scenes can be recorded. However, this solution would be prohibitively expensive in terms of data saving and processing, camera synchronization and calibration, and, most importantly, cost. Consequently, it is more realistic to build a 2D camera array system with lower camera density. Nevertheless, such a sparse 2D camera array device can only capture static or dynamic 4D Sparsely-Sampled Light Fields (SSLFs). How to reconstruct 4D DSLFs from the real-world

4D SSLFs captured by this system using the scene representations of the Image-Based Rendering (IBR) techniques [SCK07; CSN07; SKC03; KHP01] is the main focus of this thesis. In addition, to further reduce the camera redundancy of the sparse 2D camera array, a novel 1D large-scale light field capture system is developed by the Multimedia Information Processing (MIP) group of Kiel University as shown in Figure 3.1. Specifically, this large-scale 1D-grid horizontal-parallax camera array integrates 24 RGB cameras and a stepper motor that precisely controls the movement of the multi-camera rig in both horizontal and vertical directions. Moreover, this system integrates two Microsoft Kinect V2 cameras [CGM+16; YZD+15] for perceiving the depth information of the real-world light field scenes. Therefore, the aforementioned 4D DSLF reconstruction problem can potentially be solved by the classic Depth-Image-Based Rendering (DIBR) approaches [ZZY+13; ZDW10; NDP09; FDP06; Feh04].

1.2 Objectives

The raw data captured by the movable 1D large-scale light field acquisition system introduced in the previous section are not standard SSLFs, because the image planes of all the RGB cameras of this movable system are not coplanar, while the DSLF reconstruction methods to be presented in this thesis rely on standard SSLF input data. Therefore, the parameters of the orientations and positions of all the RGB cameras need to be estimated in advance. Considering that performing camera calibration for all the RGB cameras on the large-scale light field capture system at the same time is challenging, the two Kinect V2 cameras are leveraged to assist the calibration process, which is composed of two parts: RGB-Kinect calibration and Kinect registration. After employing the estimated calibration parameters to rectify the captured light field raw data, how to effectively and efficiently reconstruct an unknown desired DSLF from an input SSLF is the other challenging problem to solve.

The challenges and objectives that have been tried to be addressed in this thesis can be summarized as follows:

(i) **Camera calibration**. In order to produce standard 3D or 4D real-world SSLF data from the light field raw data captured by the novel

1D large-scale light field acquisition system, the rigid transformation from the coordinate system of each RGB camera to the world coordinate frame requires to be estimated. The challenging camera calibration problem of our light field acquisition system can be decomposed into two sub-problems: RGB-Kinect calibration and Kinect registration. Specifically, in the RGB-Kinect calibration sub-problem, we study how to approximate the camera parameters of one Kinect V2 and its nearest 12 neighboring RGB cameras, and in the Kinect registration sub-problem, we investigate how to calibrate the two Kinect V2 cameras with a large displacement, so that the extrinsic parameters of all the RGB cameras can be represented using the same coordinate system.

(ii) **Light field reconstruction**. How to reconstruct an unknown target 4D DSLF from an input 4D SSLF is also a challenging problem. The state-of-the-art techniques from the areas of computer vision, deep learning and signal processing, e. g., video frame interpolation, optical flow, Convolutional Neural Network (CNN) and Shearlet Transform (ST), are employed to tackle this problem.

1.3 Contributions

The contributions in this thesis are mainly introduced in academic publications presented in Chapter 6. As the first author concerning the listed publications, I am responsible for the main concepts, ideas, realization of software solutions, result analysis and presentation. The co-authors have contributed to the ideas and analysis. In the following all publications are presented in chronological order:

- **Publication 1**: "A Linear Method for Recovering the Depth of Ultra HD Cameras Using a Kinect V2 Sensor", Yuan Gao, Matthias Ziegler, Frederik Zilly, Sandro Esquivel and Reinhard Koch, 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), Nagoya, Japan, 8-12 May 2017, pages 494-497. (Section 6.1)
- **Publication 2**: "A Novel Kinect v2 Registration Method for Large-Displacement Environments Using Camera and Scene Constraints",

Yuan Gao, Sandro Esquivel, Reinhard Koch, Matthias Ziegler, Frederik Zilly and Joachim Keinert, 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17-20 Sept. 2017, pages 997-1001. (Section 6.2)

- **Publication 3**: "A Novel Self-Calibration Method for a Stereo-ToF System Using a Kinect V2 and Two 4K GoPro Cameras", Yuan Gao, Sandro Esquivel, Reinhard Koch and Joachim Keinert, 2017 International Conference on 3D Vision (3DV), Qingdao, China, 10-12 Oct. 2017, pages 21-28. (Section 6.3)

- **Publication 4**: "Parallax View Generation for Static Scenes Using Parallax Interpolation Adaptive Separable Convolution", Yuan Gao and Reinhard Koch, 2018 IEEE International Conference on Multimedia & Expo (ICME) Workshops, San Diego, CA, USA, 23-27 July 2018, pages 1-4. (**won the "1st Place" Award of ICME 2018 Grand Challenge on DSLF Reconstruction**, Section 6.4)

- **Publication 5**: "A Novel Kinect V2 Registration Method Using Color and Deep Geometry Descriptors", Yuan Gao, Tim Michels and Reinhard Koch, 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3-7 Sept. 2018, pages 201-205. (Section 6.5)

- **Publication 6**: "MAST: Mask-Accelerated Shearlet Transform for Densely -Sampled Light Field Reconstruction", Yuan Gao, Robert Bregovic, Atanas Gotchev and Reinhard Koch, 2019 IEEE International Conference on Multimedia and Expo (ICME), Shanghai, China, 8-12 July 2019, pages 187-192. (Section 6.6)

- **Publication 7**: "Light Field Reconstruction Using Shearlet Transform in TensorFlow", Yuan Gao, Reinhard Koch, Robert Bregovic and Atanas Gotchev, 2019 IEEE International Conference on Multimedia & Expo (ICME) Workshops, Shanghai, China, 8-12 July 2019, pages 612-612. (**won the Best Demo Award of ICME 2019**, Section 6.7)

- **Publication 8**: "IEST: Interpolation-Enhanced Shearlet Transform for Light Field Reconstruction Using Adaptive Separable Convolution", Yuan Gao, Reinhard Koch, Robert Bregovic and Atanas Gotchev, 2019

27th European Signal Processing Conference (EUSIPCO), A Coruña, Spain, 2-6, Sept. 2019, pages 1-5. (Section 6.8)

- ▻ **Publication 9**: "FAST: Flow-Assisted Shearlet Transform for Densely-Sampled Light Field Reconstruction", Yuan Gao, Reinhard Koch, Robert Bregovic and Atanas Gotchev, 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22-25 Sept. 2019, pages 3741-3745. (**Top 10% Papers of ICIP 2019**, Section 6.9)

- ▻ **Publication 10**: "DRST: Deep Residual Shearlet Transform for Densely-Sampled Light Field Reconstruction", Yuan Gao, Robert Bregovic, Reinhard Koch and Atanas Gotchev, arXiv preprint arXiv:2003.08865. (Section 6.10)

Publication 1 : A Linear Method for Recovering the Depth of Ultra HD Cameras Using a Kinect V2 Sensor

In this paper, we take full advantage of the Time-of-Flight (ToF) sensor [ZMD+16; HHE+16; GTK+13] in a Kinect V2 camera to map its depth information to an Ultra HD resolution camera. To this end, a linear least squares method is proposed. Specifically, a regular 2D checkerboard is employed to find corresponding points between the Kinect V2 sensor and the Ultra HD camera. Then, the rigid transformation between these two cameras is solved by the least squares method. Furthermore, a non-linear coarse-to-fine solution is also explored and compared with the linear one. Experimental results demonstrate the effectiveness and efficiency of the proposed linear method, which performs better than the non-linear approach.

Publication 2 : A Novel Kinect v2 Registration Method for Large-Displacement Environments Using Camera and Scene Constraints

In this publication, a novel coarse-to-fine calibration method using camera and scene constraints is proposed to solve the Kinect V2 registration problem in a large-displacement environment. To be precise, an off-the-shelf

feature detector is utilized to find constraints in the scene. The homography and fundamental matrices are employed to construct constraints in the cameras. The coarse estimation is composed of feature point detection, coarse matching, match filtering, and least-squares fitting steps. The estimation refinement consists of an ICP-based point cloud registration algorithm. Experimental results show that the fundamental matrix-based coarse-to-fine registration method outperforms the checkerboard-based coarse-to-fine registration approach on our movable multi-camera light field acquisition system having two Kinect V2 sensors in a large-displacement environment.

Publication 3 : A Novel Self-Calibration Method for a Stereo -ToF System Using a Kinect V2 and Two 4K GoPro Cameras

In this paper, we propose a depth correction step, a stereo-ToF calibration method and a depth fusion strategy to solve the two challenging problems of the multi-camera rig, i. e., the stereo-ToF calibration and fusion in 4K resolution. In particular, the depth correction step increases the depth accuracy of the Kinect V2 camera. The stereo-ToF calibration method is based on the reliable point pairs, which are detected by an off-the-shelf feature point detector and filtered using geometric constraints in the cameras and scene. Besides, the camera rotation matrix can be linearly approximated because the Kinect V2 and the GoPro cameras have similar orientations. The depth fusion strategy exploits the rigid transformation result of the stereo-ToF calibration method to fuse the depth information from the stereo matching method and ToF sensor at the pixel level. Experimental results demonstrate the effectiveness of the proposed depth correction step, stereo-ToF calibration method and depth fusion strategy.

Publication 4 : Parallax View Generation for Static Scenes Using Parallax Interpolation Adaptive Separable Convolution

In this publication, a novel parallax view synthesis method, which is based on the spatially-adaptive Separable Convolution (SepConv) [NML17b], is proposed to solve the DSLF reconstruction problem for an input SSLF

only containing parallax views for a static scene. Specifically, the proposed parallax view generation approach, Parallax-Interpolation Adaptive Separable Convolution (PIASC), leverages a fine-tuning strategy to enhance the convolution kernels of SepConv with a consideration of the motion coherence of static objects in a parallax-view capture system. The PIASC method is evaluated on all the three development datasets of ICME 2018 grand challenge on DSLF reconstruction [VSB+18] and further compared with SepConv. Experimental results demonstrate the effectiveness of the proposed PIASC and its superiority over SepConv for DSLF reconstruction of static scenes.

Publication 5 : A Novel Kinect V2 Registration Method Using Color and Deep Geometry Descriptors

In this paper, a novel camera calibration method for Kinect V2 sensors using local color and geometry information is proposed to solve the registration problem of two Kinect V2 cameras. Specifically, an off-the-shelf feature detector is used for detecting interest points and describing local color information for them. Afterwards, a CNN-based 3D descriptor, 3DMatch [ZSN+17], is utilized to describe local geometry information for these interest points. Both color and geometry descriptors are employed to estimate an initial rough rigid transformation between two Kinect V2 cameras, which can then be refined by an optional estimation refinement step if necessary. Experimental results prove the effectiveness of the proposed method by comparing it with baseline approaches.

Publication 6 : MAST: Mask-Accelerated Shearlet Transform for Densely-Sampled Light Field Reconstruction

The ST-based DSLF reconstruction approach [VBG18; VBG17; VBG15] is extremely effective in reconstructing a densely-sampled Epipolar-Plane Image (EPI) from a sparsely-sampled EPI with a large disparity range [GKB+19b]. This algorithm typically requires one to estimate the disparity range of the sparsely-sampled EPI to construct a suitable specifically-tailored universal shearlet system [VBG18; GK14]. Besides, the sparsely-sampled EPI also needs the disparity information for shearing and zero

padding in order to be correctly processed by this elaborately-designed shearlet system. Moreover, for DSLF reconstruction from SSLFs with large disparity ranges, this algorithm tends to be time-consuming due to the high number of iterations of its iterative thresholding algorithm. Therefore, in this paper, a novel ST-based coarse-to-fine DSLF reconstruction method, referred to as Mask-Accelerated Shearlet Transform (MAST), is proposed to address these two problems. The presented MAST method takes full advantage of a state-of-the-art learning-based optical flow estimation approach, i. e., FlowNet2 [IMS+17], to estimate the disparities of the whole SSLF for resolving the first problem. In addition, the estimated disparities are also used to roughly restore a densely-sampled EPI from a sparsely-sampled EPI via inverse warping. The iterative estimation refinement algorithm in ST convergences faster by means of an elaborately-designed soft mask for the coarsely-inpainted densely-sampled EPI, thus tackling the second problem. Experimental results demonstrate the superior performance of MAST over the other state-of-the-art DSLF reconstruction methods on nine challenging horizontal-parallax real-world light field datasets with disparity ranges up to 35 pixels.

Publication 7 : Light Field Reconstruction Using Shearlet Transform in TensorFlow

This demo paper presents a comprehensive implementation of ST for light field reconstruction using one of the most popular machine learning libraries, i. e., TensorFlow. The flexible architecture of TensorFlow allows for the easy deployment of ST across different platforms (CPUs, GPUs, TPUs) running varying operating systems with high efficiency and accuracy.

Publication 8 : IEST: Interpolation-Enhanced Shearlet Transform for Light Field Reconstruction Using Adaptive Separable Convolution

In this paper, a novel method, referred to as Interpolation-Enhanced Shearlet Transform (IEST), is proposed to address the challenging light field reconstruction problem for the cases of moderate disparity range (8 - 16

pixels) and large disparity range (> 16 pixels). The proposed IEST method fully leverages the advantages of both ST and SepConv in a coarse-to-fine manner to reconstruct a target light field from a horizontal-parallax SSLF with a moderate or large disparity range. Specifically, ST is employed to reconstruct the target light field $\mathcal{D}$ from an input SSLF $\mathcal{S}$, so that the missing parallax views in $\mathcal{D}\backslash\mathcal{S}$ are coarsely estimated. Two elaborately-designed parallax view refinement strategies, corresponding to different interpolation rates $\delta \in \{8, 16\}$, are then applied to the coarsely-estimated $\mathcal{D}$ in a recursive manner. Experimental results indicate that IEST outperforms all the other state-of-the-art methods on nine challenging horizontal-parallax evaluation SSLF datasets for both the moderate and large disparity ranges.

Publication 9 : FAST: Flow-Assisted Shearlet Transform for Densely-Sampled Light Field Reconstruction

As explained above, when using ST for DSLF reconstruction, the disparity information of the input SSLF is required to be obtained in advance for (i) constructing a decent shearlet system; (ii) pre-shearing the SSLF in order to eliminate the minimum disparity of it. To tackle the disparity estimation problem, a state-of-the-art optical flow algorithm, i. e., PWC-Net [SYL+18], is exploited to estimate the bidirectional disparity maps between adjacent views in the SSLF. In addition to assisting the DSLF reconstruction of using ST, the estimated bidirectional disparity maps can also be used to perform DSLF reconstruction via novel view synthesis using image warping and blending techniques. However, due to the occlusion and errors in the estimated disparity maps, this disparity-based solution to DSLF reconstruction may not produce visually pleasing results. Therefore, to improve the performance of both ST-based and disparity-based DSLF reconstruction methods, a novel learning-based approach, referred to as Flow-Assisted Shearlet Transform (FAST), is proposed in this paper. The FAST method makes full use of the bidirectional disparity maps predicted by PWC-Net and the DSLF recovered by ST to better reconstruct the target DSLF via two deep CNNs, i. e., Disparity Refinement Network (DRN) and View Synthesis Network (VSN). Additionally, the proposed FAST is fully convolutional and end-to-end trainable. Experimental results on nine evaluation DSLF sub-datasets demonstrate the effectiveness of FAST for

reconstructing DSLFs from SSLFs with large disparity ranges.

Publication 10 : DRST: Deep Residual Shearlet Transform for Densely-Sampled Light Field Reconstruction

In this paper, a novel learning-based approach, referred to as Deep Residual Shearlet Transform (DRST), is proposed to address the fundamental speed issue of ST. In particular, DRST performs shearlet coefficient reconstruction in shearlet domain for an input sparsely-sampled EPI by means of a deep CNN, which is composed of a residual learning strategy and an encoder-decoder network that predicts the residuals of the shearlet coefficients. In other words, the learning-based DRST is essentially a regression model, which maps the shearlet coefficients of an input sparsely-sampled EPI to unobserved shearlet coefficients of the target densely-sampled EPI. The reconstructed shearlet coefficients in shearlet domain are then transformed back into image domain to produce a corresponding inpainted densely-sampled EPI. Finally, a target DSLF can be reconstructed by repeating this EPI reconstruction process on all the sparsely-sampled EPIs of the input SSLF. Besides, the network of DRST is fully convolutional and end-to-end trainable. Considering the difficulty of acquiring ground-truth DSLFs, the training of DRST is performed solely on SSLF data via a self-supervised manner [JT20]. The synthetic SSLF data are used for training because the ground-truth disparity information, which is beneficial for the shearlet construction, pre- and post-shearing steps of DRST, can be provided by using the state-of-the-art 3D computer graphics softwares.

The key contributions of this paper are as follows.

- We propose a learning-based DRST method that achieves better DSLF reconstruction performance than the non-learning-based ST algorithm on three evaluation datasets composed of real-world horizontal-parallax light fields with different moderate disparity ranges (8 - 16 pixels);
- The network of DRST is trained solely on synthetic SSLF data in a self-supervised fashion by means of the elaborately-designed masks. To our best knowledge, this is the first work to investigate learning-based DSLF reconstruction with *only exploiting synthetic SSLFs as training data*;
- The proposed deep learning-based DRST is more time-efficient than the

classical model-based sparse regularization using ST. Specifically, DRST provides a 2.4x speedup over ST, at least;

- Experimental results on three challenging real-world light field evaluation datasets show that the performance of DRST is better than, or at least comparable to, the other state-of-the-art DSLF reconstruction methods.

1.4 Outline

This thesis is organized in six chapters.

- In Chapter 1, we start with the motivation and describe the objectives and contributions of this thesis;
- Chapter 2 outlines the introduction to the background and related work of large-scale light field acquisition, calibration and reconstruction;
- Chapter 3 describes the camera model, RGB-Kinect calibration and Kinect registration for our large-scale 1D light field acquisition system;
- We investigate in Chapter 4 how to reconstruct a DSLF from a SSLF from three perspectives, i. e., novel view synthesis, EPI inpainting and the fusion of them;
- Chapter 5 concludes and summarizes the presented work followed by a discussion of the future research;
- Finally, Chapter 6 lists the academic publications which build the basis for the research work of this thesis.

Chapter 2

Background

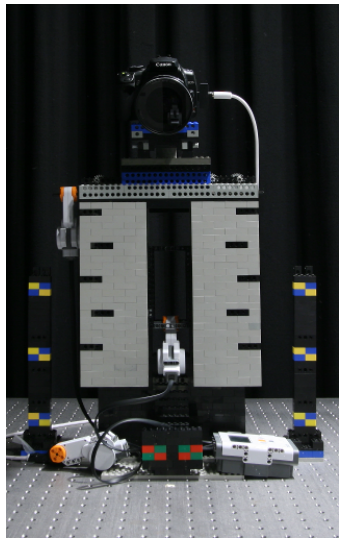
This chapter will review and discuss the background and recent developments related to large-scale light field acquisition [UWH+03], calibration [XMN+15; VWJ+04; KHP+99] and reconstruction [Vag20; VBG20; BSV+19].

2.1 Light field acquisition

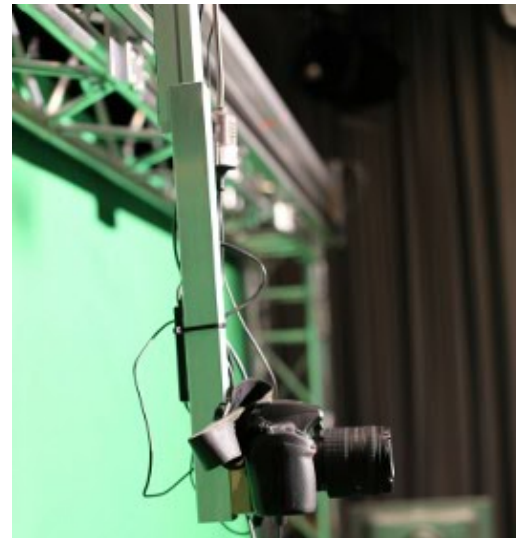
Depending on the camera arrangement of single or multiple RGB cameras for capturing light fields in real-world environments, the large-scale light field acquisition systems can be classified into three categories, i. e., single-camera light field gantry, 2D light field array and 1D light field array.

2.1.1 Single-camera light field gantry

A single-camera light field gantry from Stanford University is shown in Figure 2.1 (a). It can be seen that this gantry is built using Lego Mindstorms with integrating a Digital Single-Lens Reflex camera (DSLR) that can be moved horizontally and vertically [VA08]. There are 13 different light field scenes, involving translucency and complex specular geometry, captured by this Lego gantry; as a result, 13 relatively-dense 4D light fields with the same angular resolution of 17×17 but different spatial resolutions are produced. Recently, Fraunhofer IIS also builds a single-camera light field gantry presented in Figure 2.1 (b) [SBZ+18; ZVK+17]. Different from the Stanford one, this gantry uses a high-precision cantilever axes to capture large-scale static 4D light fields. Specifically, the DSLR can be translated by up to 4 m horizontally and 0.5 m vertically with a precision error of 80 µm. The High Density Camera Array (HDCA) dataset [ZVK+17] captured by this system contains nine real-world 4D light fields in 4K



(a) Stanford Lego gantry
(Source: [VA08])



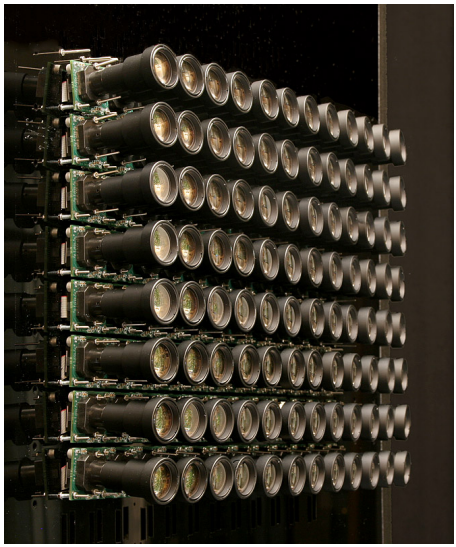
(b) Fraunhofer IIS gantry
(Source: [LQF+16])

Figure 2.1. Single-camera light field gantries for capturing static 4D light fields.

spatial resolution, which can be used as an evaluation dataset for the performance comparison of different light field reconstruction methods presented in Chapter 4. The advantage of these two single-camera light field gantries is that they are capable of capturing relatively-dense static 4D or 3D light fields. However, such systems can hardly capture dynamic light fields, i. e., light fields of dynamic scenes.

2.1.2 2D light field array

As a pioneer of developing large-scale light field acquisition systems, Wilburn et al. build three types of 2D large-scale camera arrays [WJV+05], of which one is illustrated in Figure 2.2 (a). The large-scale light field acquisition systems built by them have several light field applications, including high-resolution and high-dynamic range video capture, high-speed video capture, spatiotemporal view interpolation and synthetic aperture image



(a) Stanford multi-camera array
(Source: [WJV+05])

(b) Lytro immerge 2.0
(Original image courtesy of Lytro)

Figure 2.2. 2D light field arrays for capturing dynamic 4D light fields.

generation. Recently, Lytro builds a novel large-scale 2D light field camera rig, referred to as Immerge 2.0, for high-end VR production [MSG17]. As shown in Figure 2.2 (b), this rig, consisting of 95 individual cameras, is able to capture 360-degree content in three spins. The advantage of these two 2D light field arrays is that both of them can capture full-parallax static and dynamic light fields. However, in order to capture large-scale light fields from varying real-world scenes, the sizes of these 2D light field arrays tend to be huge and the costs of them tend to be high. In addition, how to save and process the large amount of light field data captured by them in real time is challenging.

2.1.3 1D light field array

Different from the above 2D light field arrays that capture full-parallax light fields, an 1D light field array captures horizontal- or vertical-parallax

Figure 2.3. 100 - camera system, a large-scale 1D light field array for capturing dynamic 3D light fields. (Source: [Tan06])

light fields. One of the most famous 1D light field arrays is "100 - camera system". It is developed by Nagoya University to capture large-scale light field scenes for real-time rendering applications in free viewpoint television (FTV) systems [Tan06; FMT+06]. As shown in Figure 2.3, this system is composed of 100 RGB cameras, one host-server PC and 100 client PCs (called "nodes") connected to each camera. Besides, the host-server PC generates synchronization signals and distributes them to all the nodes, each of which is able to capture high-resolution (maximum 1280×960 pixels) videos at 30 frames/s. Compared to the setups of the 2D light field arrays introduced above, a 1D light field array will be cheaper and easier to handle w.r.t. data saving and processing if the number of cameras reduces from $n \times n$ to n. Another advantage of 1D light field array is that the captured 3D light field data can be directly fed to glassless, holographic 3D light field display systems, such as HoloVizio developed by Holografika [MBB08; Bal06; ABF+06]. However, for rendering tasks on VR devices that require the full-parallax information of the scene, the horizontal-parallax-only light field data captured by the 1D light field array may be inadequate. To overcome this limitation, a novel movable 1D light field array for capturing full-parallax light fields is designed by us and will be elaborated in Section 3.1.

2.2 Camera calibration

Since our novel 1D large-scale light field acquisition system has multiple RGB and depth cameras. The related work of camera calibration for these cameras is introduced in this section.

2.2.1 RGB-depth sensor calibration

The Perspective-n-Point (PnP) is one of the most common solutions to solve the calibration problem between a depth sensor and an RGB camera. The PnP problem is first described in [FB81], which stands for the problem of how to estimate the camera pose of a calibrated camera using n known 3D reference points in the world coordinate frame and their corresponding 2D points on the camera image plane of this calibrated camera. The solutions to the PnP problem can be classified into two categories:

(i) **Iterative methods**. Lu et al. minimize an object-space collinearity error for computing orthogonal rotation matrices, which is proven to be globally convergent [LHM00]. Zhang proposes a closed-form solution for estimating the camera intrinsic and extrinsic parameters, which can be then refined by leveraging the Levenberg-Marquardt algorithm [Zha00].

(ii) **Non-iterative methods**. Lepetit et al. express the non-iterative solution to the PnP problem as a vector standing for a weighted sum of the null eigenvectors and their method achieves the computational complexity growing linearly with n [LMF09]. Li et al. also present an $O(n)$ solution by estimating the coordinates of two special end points [LXX12].

2.2.2 Depth sensor registration

To address the registration problem of multiple depth cameras, several methods have been proposed with using calibration objects. Afzal et al. propose an RGB-D multi-view system calibration method, i. e., BAICP+, which combines Bundle Adjustment (BA) [Zac14; ASS+10; LA09; TMH+00]

and Iterative Closest Point (ICP) [BM92] into a single minimization framework [AAF+14]. The corners of a checkerboard are detected in the BA part of BAICP+. Kowalski et al. present a coarse-to-fine solution to the problem of multi-Kinect V2 calibration, where a planar marker is used for the rough estimation of camera poses, which is later refined by an ICP algorithm [KND15]. Soleimani et al. employ three double-sided checkerboards placed at varying depths to perform an automatic calibration process of two opposing Kinect V2 cameras [SMD+16]. Córdova-Esparza et al. introduce a calibration tool for multiple Kinect V2 sensors using a 1D calibration object, i. e., a wand, which has three collinear points [CTJ+17].

2.3 Light field reconstruction

Originally defined in [VBG15], DSLF has a wide range of applications, including depth estimation, super-resolution, synthetic aperture imaging, and visualization on 3DTV, VR and holographic devices [WMJ+17]. As explained in Section 1.1, due to the hardware limitations of modern light field acquisition systems, the goal of directly capturing a desired DSLF for a real-world scene can hardly be achieved by using these systems; however, a SSLF with a moderate (8 - 16 pixels) or large disparity range (> 16 pixels) for the same real-world scene is easy to capture by most of them. Therefore, performing an effective and efficient DSLF reconstruction on the real-world SSLFs with moderate or large disparity ranges is the best way to compensate for the hardware limitations of these light field acquisition systems. The related work to light field reconstruction can be categorized into two types: video frame synthesis and light field novel view synthesis, which are introduced in the following two subsections, respectively.

2.3.1 Video frame synthesis

Since real-world SSLFs can be converted into videos captured by virtual cameras, the video frame synthesis approaches can be leveraged to reconstruct the target DSLFs from these SSLFs. Typically, the video frame synthesis methods can be classified into three groups: (i) phase-based

methods, (ii) flow-based methods and (iii) kernel-based methods, which are described as follows.

(i) **Phase-based methods**. Meyer et al. propose a phase-based image synthesis approach to synthesize in-between images [MWZ+15]. Recently, Meyer et al. apply the steerable pyramid filters [SF95] to decompose the input two consecutive video frames [MDM+18]. Their decompositions, consisting of amplitudes, phases and low-pass residuals, are fed to a decoder-only neural network, i. e., PhaseNet, to predict the corresponding decomposition of the intermediate frame in order to fulfill image reconstruction. Visually preferable results are achieved by this method in challenging scenarios containing lighting changes or motion blur.

(ii) **Flow-based methods**. Liu et al. propose an end-to-end deep network, i. e., deep voxel flow, to synthesize a novel video frame in either interpolation or extrapolation with sharp results [LYT+17]. Niklaus et al. fully leverage a state-of-the-art optical flow algorithm, i. e., PWC-Net [SYL+18], to estimate bidirectional flow between two consecutive input video frames [NL18]. The estimated bidirectional flow is then used to pre-warp the input video frames together with their corresponding per-pixel context maps extracted by a pre-trained neural network [HZR+16]. Finally, all these pieces of the pre-warped data are fed to a video frame synthesis network, i. e., an adapted GridNet [FEF+17], to interpolate an intermediate video frame at a desired temporal position. Jiang et al. also estimate bidirectional optical flow between two consecutive input video frames via a flow computation CNN [JSJ+18]. The estimated optical flow is then refined by a flow interpolation CNN, which additionally predicts a soft visibility map. Both the refined optical flow and predicted soft visibility map are utilized to interpolate an intermediate video frame at any arbitrary time step via warping and fusion. More recently, Xu et al. propose a quadratic video interpolation approach that exploits the acceleration information for acceleration-aware motion estimation and high-quality frame synthesis [XLS+19; LXP+19; NST+19].

(iii) **Kernel-based methods**. Niklaus et al. employ a deep fully CNN to estimate pixel-wise spatially-adaptive 2D convolution kernels, which are applied to the two consecutive input video frames to synthesize an intermediate one [NML17a]. However, for each image pixel, this method predicts a $\varrho \times \varrho$ ($\varrho = 41$) convolution kernel, which will be prohibitively expensive in terms of memory requirement if the resolution of the two input video frames is high. To tackle this problem, Niklaus et al. propose a spatially-adaptive separable convolution approach, i. e., SepConv, to approximate each of the 2D convolution kernels using a pair of 1D kernels, thus reducing the number of kernel parameters from ϱ^2 to 2ϱ for each 2D convolution kernel [NML17b]. More recently, Bao et al. propose the motion estimation and motion compensation driven neural network [BLZ+19] and depth-aware video frame interpolation approach [BLM+19], which integrate both interpolation kernels and optical flow, for video frame synthesis and enhancement.

2.3.2 Light field novel view synthesis

The DSLF reconstruction problem can also be solved by light field novel view synthesis methods, which can be categorized into three groups: (i) angular resolution enhancement, (ii) Multiplane image (MPI) and (iii) neural rendering. More details about these three groups are introduced as follows.

(i) **Angular resolution enhancement**. Kalantari et al. propose a deep learning-based view synthesis approach, which is composed of disparity and color estimators, for synthesizing novel views from a sparse set of sub-aperture images of a micro-lens array-based consumer light field camera [KWR16]. Wu et al. present a blur-restoration-deblur framework for EPI interpolation to reconstruct light fields [WZW+17]. A residual network with three convolution layers is utilized to restore the angular detail of a blurred and upsampled EPI. However, due to the limitation in the blurring kernel size and bicubic interpolation, this method can only handle SSLF data with very small disparity ranges (up to 5 pixels). Vagharshakyan

et al. reconstruct DSLFs from SSLFs by taking full advantage of EPI sparsification in shearlet domain [VBG18; VBG17; VBG15]. Their method is referred to as ST and demonstrated to be effective in reconstructing Lambertian scenes and non-Lambertian scenes containing semi-transparent objects. The EPI sparsification is essentially an iterative hard thresholding algorithm in shearlet domain, which requires dozens of iterations of domain transformations between image domain and shearlet domain. Therefore, ST tends to be slow for input SSLFs with high spatial or angular resolution. Yeung et al. design an end-to-end 4D convolutional light field reconstruction network consisting of view synthesis and view refinement phases for fast light field reconstruction on an input SSLF [YHC+18]. Wang et al. also propose an end-to-end learning framework for fast light field reconstruction [WLW+18]. Their network includes two 2D strided convolutions for the interpolation of stacked sparsely-sampled EPIs and two detail-restoration 3D CNNs for restoring high-frequency details of these interpolated EPI volumes. In conclusion, all the above deep learning-based approaches are based on supervised learning, which requires a lot of ground-truth training data. In terms of the problem of DSLF reconstruction, it is extremely difficult to capture ground-truth DSLF training data as introduced in the beginning of this thesis (Section 1.1). To resolve this problem, Gao et al. propose a novel self-supervised DSLF reconstruction approach, referred to as CycleST, of which the network can be trained solely on synthetic SSLF data [GBG20].

(ii) **Multiplane image (MPI)**. Zhou et al. propose a layered scene representation, i. e., MPI, and a learning framework for stereo magnification on narrow-baseline stereo image pairs [ZTF+18]. The MPI representation has an advantage that it is extremely suitable for rendering high-quality and high-fidelity novels views from input SSLFs in real time. Mildenhall et al. fully leverage the MPI representation to synthesize novel views in real time by blending adjacent local light fields [MSO+19]. This method is demonstrated to be effective for rendering challenging non-Lambertian effects. The maximum disparity between input view samples that can be effectively handled by it

is up to 64 pixels. Flynn et al. propose a novel method, DeepView, which employs the learned gradient decent and MPI scene representation, for real-time high-quality view synthesis [FBD+19; DFB+19]. Li et al. propose a neural IBR approach using DeepMPI representation, i. e., an extension of MPI, to synthesize arbitrary views from crowdsourced images [LXD+20]. Although the MPI representation facilitates real-time light field rendering, the process of generating MPIs is time- and storage-intensive. In particular, it may take a few dozens of seconds to infer a MPI from a high-resolution image on a current high-end GPU. In addition, such a MPI is represented by a large 3D voxel grid, resulting in enormous storage requirements.

(iii) **Neural rendering**. Apart from the aforementioned angular resolution enhancement and MPI solutions, the neural rendering techniques can also be applied to light field novel view synthesis [TFT+20]. Mildenhall et al. propose to represent a static scene using a 5D neural radiance field representation, i. e., a continuous 5D volumetric scene function that takes the 3D spatial location and 2D viewing direction as input and predicts the volume density and view-dependent emitted radiance [MST+20]. Their IBR method using this representation is capable of synthesizing high-quality and photorealistic novel views for photometrically static scenes containing complex geometry and materials. Martin-Brualla et al. adapt the neural radiance field representation to perform high-fidelity light field rendering from unstructured collections of in-the-wild photographs [MRS+20]. Overall, the neural radiance field-based methods are much slower than the MPI-based methods and can hardly be applied to high-resolution light field rendering in real time.

Chapter 3

Camera Calibration

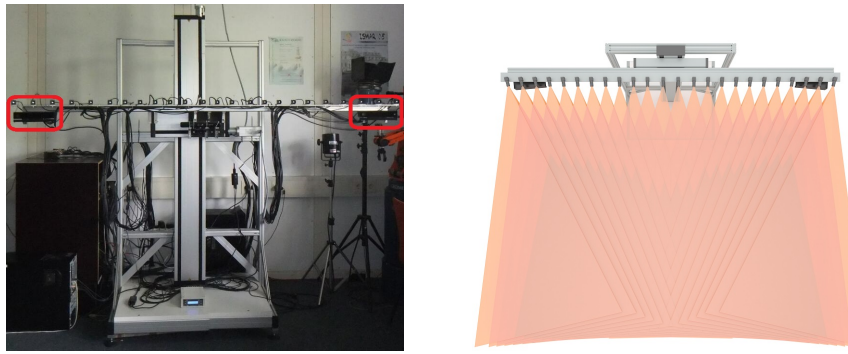
In order to capture large-scale light fields from real-world scenes, the MIP group has developed a novel 1D large-scale light field acquisition system. In this chapter, we focus on the design and calibration of this system. In particular, the specification of this system and associated camera calibration and light field reconstruction problems will be introduced in the next section. The details of our solutions to the camera calibration problems of this system will be presented in the rest three sections.

3.1 Large-scale light field acquisition system

The novel 1D large-scale light field acquisition system designed by the MIP group for capturing 3D and 4D real-world SSLFs is illustrated in Figure 3.1. As can be seen from both (a) and (b), this system is composed of 24 RGB cameras, two Kinect V2 sensors, a microstep controller and a hardware trigger [EGM+16]. Specifically,

- the 24 RGB cameras having large baselines[1] ($\approx$ 11 cm) are exploited to capture either static or dynamic SSLFs in real-world environments;
- the two Kinect V2 sensors having a large displacement ($\approx$ 2.4 m) are leveraged to perceive the depth information of real-world scenes;
- the microstep controller is in charge of moving the rig horizontally (up to $\approx$ 25 cm) and vertically (up to $\approx$ 2 m);
- and the hardware trigger is utilized to synchronize all the 24 RGB cameras.

[1] Here, a baseline stands for the distance between any two neighboring RGB cameras.



(a) Front view (b) Virtual vertical view

Figure 3.1. Movable multi-camera rig for capturing large-scale static and dynamic light fields. Red blocks in (a) indicate the positions of two Kinect V2 sensors. The positions, orientations and FoV of all the 24 RGB cameras are illustrated in (b). (Source: [GEK+17b])

Besides, the color image resolution and Field of View (FoV) of each RGB camera are 1280×1024 pixels and $31° \times 25°$, respectively; however, the ToF sensor in the Kinect V2 camera has a much lower depth image resolution (512×424 pixels) and a much higher FoV ($70° \times 60°$). When using such a setup for capturing large-scale static and dynamic light fields, we encounter three challenging problems:

(i) The resolution of any RGB camera is around 2.5x larger than that of the ToF sensor of the Kinect V2 camera. In addition, there is a large FoV difference between the RGB cameras and the ToF sensors. How to recover the position and orientation information of each RGB camera in the world by only using Kinect V2 cameras is challenging;

(ii) The horizontal displacement between the two Kinect V2 cameras on the multi-camera rig is around 2.4 m. How to register these two Kinect V2 cameras with such a large displacement is challenging;

(iii) The 3D or 4D light fields captured by the movable 1D large-scale light field acquisition system are sparsely-sampled. How to reconstruct DSLFs from these SSLFs is challenging.

To address the challenging problems (i) and (ii) that are related to camera calibration, we propose two types of methods, i. e., the RGB-Kinect calibration of one Kinect V2 camera and multiple RGB cameras in Section 3.3 and Kinect registration of the two Kinect V2 sensors in Section 3.4. These methods rely on the basic camera model introduced in Section 3.2. Besides, how to tackle the challenging problem (iii) related to light field reconstruction will be elaborated in Chapter 4.

3.2 Camera model

A camera model describes the mapping between the objects in the 3D world and their projection onto the 2D images captured by a projective camera. The methods for solving the camera calibration and registration problems of our 1D large-scale light field acquisition system are based on the classic finite projective camera model [HZ03], which is illustrated in Figure 3.2. Typically, this model is composed of three components, i. e., rigid transformation, intrinsic matrix and lens distortion, which are presented in the following three subsections.

3.2.1 Rigid transformation

Let a homogeneous 4-vector $\boldsymbol{P}^w = \begin{pmatrix} P_x^w, & P_y^w, & P_z^w, & 1 \end{pmatrix}^{\mathrm{T}} \in \mathbb{P}^3$ denote a 3D point in the world coordinate frame. The center of a camera in the world coordinate frame is represented by an inhomogeneous 3-vector $\tilde{\boldsymbol{C}}^w = \begin{pmatrix} C_x^w, & C_y^w, & C_z^w \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^3$. The 3D point $\boldsymbol{P}^w$ in the world coordinate frame can then be converted into a 3D point $\boldsymbol{P}^c = \begin{pmatrix} P_x, & P_y, & P_z, & 1 \end{pmatrix}^{\mathrm{T}}$ in the camera coordinate frame by using a rotation matrix $\mathbf{R} \in \mathbb{SO}(3)$ and the camera center $\tilde{\boldsymbol{C}}^w$ in the world coordinate frame as below:

$$\boldsymbol{P}^c = \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\boldsymbol{C}}^w \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{P}^w. \tag{3.2.1}$$

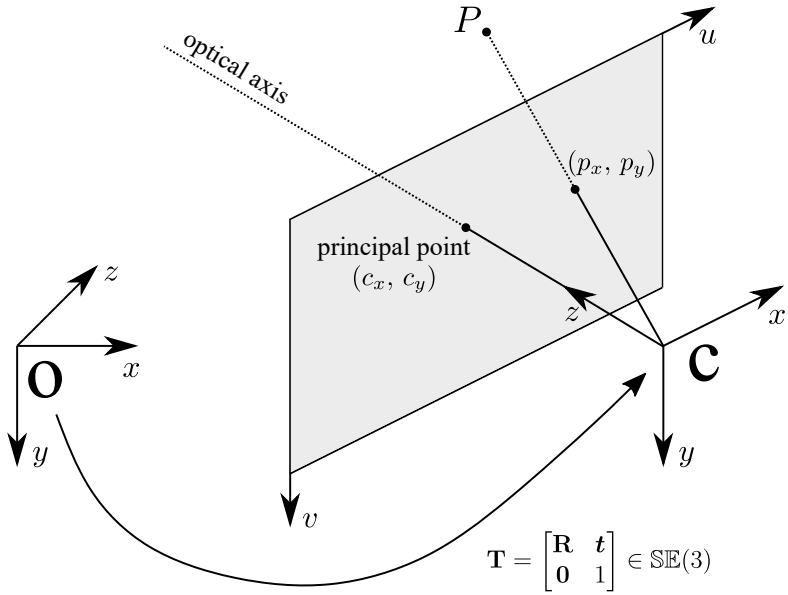


Figure 3.2. Finite projective camera model. The symbol 'O' denotes the world coordinate system. The symbol 'C' stands for the camera coordinate system. The symbol 'P' represents a 3D point that can be seen by the camera. The "uv" plane is the camera image plane. The rigid transformation from the world coordinates to the cameras coordinates is denoted by **T**.

Let the homogeneous 4-vector $\boldsymbol{P}^c$ be replaced by its inhomogeneous representation $\tilde{\boldsymbol{P}}^c = \begin{pmatrix} P_x, & P_y, & P_z \end{pmatrix}^{\mathrm{T}}$, Equation 3.2.1 can then be written as

$$\tilde{\boldsymbol{P}}^c = \mathbf{R}\left[\mathbf{I} \mid -\tilde{\boldsymbol{C}}^w\right]\boldsymbol{P}^w. \tag{3.2.2}$$

It is more common to use the rotation matrix **R** and a translation vector $\boldsymbol{t} \in \mathbb{R}^3$ to describe the rigid transformation from the world coordinate frame to the camera coordinate frame in Equation 3.2.2, i. e.,

$$\tilde{\boldsymbol{P}}^c = \left[\mathbf{R} \mid \boldsymbol{t}\right]\boldsymbol{P}^w, \tag{3.2.3}$$

where

$$\boldsymbol{t} = -\mathbf{R}\tilde{\boldsymbol{C}}^w. \tag{3.2.4}$$

The rotation matrix $\mathbf{R}$ and translation vector $\boldsymbol{t}$ in Equation 3.2.3 constitute the rigid transformation $\mathbf{T}$ from the world coordinate system to the camera coordinate system, which can be written as

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SE}(3). \tag{3.2.5}$$

Additionally, the rigid transformation $\mathbf{T}$ has 6 degrees of freedom.

3.2.2 Intrinsic matrix

The 3D point $\tilde{\boldsymbol{P}}^c \in \mathbb{R}^3$ in the camera coordinate frame is then projected onto the image plane of the camera, i. e.,

$$\begin{pmatrix} p_x \\ p_y \\ f \end{pmatrix} = \begin{bmatrix} fm_x & & c_x \\ & fm_y & c_y \\ & & f \end{bmatrix} \frac{\tilde{\boldsymbol{P}}^c}{P_z}. \tag{3.2.6}$$

Here, f is the focal length of the camera in terms of distance unit dimensions; m_x and m_y are two scaling factors representing the number of pixels per unit distance along the 'u' and 'v' axes, respectively; and (c_x, c_y) denote the coordinates of the principal point of the camera in terms of pixel dimensions. On the image plane of the camera, the 2D point represented using homogeneous coordinates, i. e., $\boldsymbol{p} = \begin{pmatrix} p_x, & p_y, & 1 \end{pmatrix}^{\mathrm{T}}$, corresponding to the 3D point $\tilde{\boldsymbol{P}}^c$ in the camera coordinate system, can then be derived from Equation 3.2.6 as below:

$$\boldsymbol{p} = \mathbf{K}\,\mathrm{proj}(\tilde{\boldsymbol{P}}^c), \tag{3.2.7}$$

where

$$\mathbf{K} = \begin{bmatrix} f_x & e & c_x \\ & f_y & c_y \\ & & 1 \end{bmatrix} \text{ and } \mathrm{proj}(\tilde{\boldsymbol{P}}^c) = \frac{\tilde{\boldsymbol{P}}^c}{P_z}. \tag{3.2.8}$$

In the above formulas, f_x $(= fm_x)$ and f_y $(= fm_y)$ are the focal lengths expressed in pixel units, the symbol 'e' is referred to as skew parameter, and the matrix $\mathbf{K}$ represents the intrinsic matrix of the camera, which has 5 degrees of freedom. Besides, $\mathrm{proj}(\cdot)$ denotes the perspective projection function, which projects an input 3D point in the camera space onto the normalized image plane of the camera. The normalized image plane is a

virtual plane of the camera, satisfying that the distance between it and the camera center is one.

3.2.3 Lens distortion

Apart from the intrinsic matrix presented above, the lens distortion is the other important component for any projective camera. This subsection mainly introduces two common types of lens distortion, i. e., radial distortion and tangential (decentering) distortion [Bro71], of which details are presented as follows.

Let a homogeneous 3-vector $\boldsymbol{p}' = \begin{pmatrix} p'_x, & p'_y, & 1 \end{pmatrix}^{\mathrm{T}}$ represent a point on the normalized image plane of the camera corresponding to the 3D point $\tilde{\boldsymbol{P}}^c$ in the camera coordinate system. From Equation 3.2.7, we get

$$\boldsymbol{p}' = \mathrm{proj}(\tilde{\boldsymbol{P}}^c) = \mathbf{K}^{-1}\boldsymbol{p}. \tag{3.2.9}$$

The above formula holds only when the lens distortion is not considered. For better calibrations of physical cameras, we need to take lens distortion into account. Lens distortion describes the mathematical relationship between the point $\boldsymbol{p}' = \mathrm{proj}(\tilde{\boldsymbol{P}}^c)$ on the normalized *undistorted* image plane and the point represented by the homogeneous 3-vector $\boldsymbol{p}'' = \begin{pmatrix} p''_x, & p''_y, & 1 \end{pmatrix}^{\mathrm{T}}$ on the normalized *distorted* image plane as below:

$$\boldsymbol{p}'' = \begin{bmatrix} \nu & & 2\lambda_1 & 2\lambda_2 & & \lambda_2 & \\ & \nu & 2\lambda_2 & & 2\lambda_1 & \lambda_1 & \\ & & & & & & 1 \end{bmatrix} \begin{pmatrix} p'_x \\ p'_y \\ p'_x p'_y \\ (p'_x)^2 \\ (p'_y)^2 \\ r^2 \\ 1 \end{pmatrix}, \tag{3.2.10}$$

where

$$\begin{aligned} r^2 &= (p'_x)^2 + (p'_y)^2, \\ \nu &= \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6}. \end{aligned} \tag{3.2.11}$$

Note that λ_1 and λ_2 in Equation 3.2.10 are the camera tangential distortion

coefficients and k_1, k_2, k_3, k_4, k_5 and k_6 in Equation 3.2.11 are the radial distortion parameters of the camera. The point $\boldsymbol{p}''$ on the normalized distorted image plane can then be mapped onto the image plane of the camera using Equation 3.2.7, i. e.,

$$\boldsymbol{p} = \mathbf{K}\boldsymbol{p}''. \tag{3.2.12}$$

The complete finite projective camera model including rigid transformation, intrinsic matrix and lens distortion can be summarized as follows:

1. Transform $\boldsymbol{P}^w \in \mathbb{P}^3$ in the world coordinate system into $\tilde{\boldsymbol{P}}^c \in \mathbb{R}^3$ in the camera coordinate system via the rigid transformation in Equation 3.2.5;
2. Convert $\tilde{\boldsymbol{P}}^c$ into $\boldsymbol{p}' = \text{proj}(\tilde{\boldsymbol{P}}^c) \in \mathbb{P}^2$ on the normalized *undistorted* image plane;
3. Map $\boldsymbol{p}'$ to $\boldsymbol{p}'' \in \mathbb{P}^2$ on the normalized *distorted* image plane using Equation 3.2.10 and Equation 3.2.11;
4. Project $\boldsymbol{p}''$ to the image plane of the camera to get the final 2D point $\boldsymbol{p} \in \mathbb{P}^2$ using Equation 3.2.12.

3.3 RGB-Kinect calibration

In this section, we propose a novel coarse-to-fine RGB-Kinect calibration approach [GEK+17a; GZZ+17], which is composed of a coarse estimation phase, where the rigid transformation from a Kinect V2 camera coordinates to each RGB camera coordinates is estimated individually, and an estimation refinement phase, where all the coarsely-estimated rigid transformations between the RGB cameras and Kinect cameras are refined in a global manner.

3.3.1 Preliminary

For the sake of clarity, in our 1D large-scale light field acquisition system, we use $\mathbb{C}_j$ ($1 \leqslant j \leqslant 24$) to denote the 24 RGB cameras, and $\mathbb{C}_A$ and $\mathbb{C}_B$ to represent the two Kinect V2 sensors, respectively. In addition, the intrinsic parameters, consisting of intrinsic matrix and lens distortion, of

all the RGB cameras are measured by a conventional checkerboard-based method [Zha00]. The intrinsic parameters of the two Kinect sensors are extracted from the factory calibration by using the Kinect for Windows SDK. The estimated and extracted intrinsic parameters are then exploited to eliminate the lens distortions of all the RGB and Kinect cameras. It should be noted that a Kinect V2 camera is composed of a ToF sensor and an RGB camera. In order to avoid ambiguity when talking about camera calibration and registration associated with the Kinect V2 camera, the ToF sensor is regarded as the default camera representing the entire Kinect V2. Moreover, the depth accuracy of a Kinect V2 camera has a constant offset of -18 mm, which has been well investigated in [WS16]. It is important to correct the depth images captured by the ToF sensors of the Kinect V2 cameras by eliminating the constant depth offset in order to get more accurate camera calibration and registration results.

With regard to our solution to RGB-Kinect calibration, the 24 RGB cameras on the multi-camera rig in Figure 3.1 are split into two groups, i. e., the left 12 RGB cameras $\mathbb{C}_j$ ($1 \leqslant j \leqslant 12$) that are close to the left Kinect camera $\mathbb{C}_A$ and the right 12 RGB cameras $\mathbb{C}_j$ ($13 \leqslant j \leqslant 24$) that are close to the right Kinect camera $\mathbb{C}_B$. There are two reasons for this process:

(i) a common size checkerboard is difficult to capture by all the 24 RGB cameras simultaneously; however, it is much easier to capture the same checkerboard by the left or right 12 RGB cameras at the same time;

(ii) the camera orientations of the RGB cameras in group one are similar to that of $\mathbb{C}_A$ and the same for group two, which facilitates the coarse estimation step of the proposed RGB-Kinect calibration method that will be described below.

Since performing RGB-Kinect calibration on the cameras of group one is the same for group two, here we only present how to calibrate the left 12 RGB cameras $\mathbb{C}_j$ ($1 \leqslant j \leqslant 12$) and the left Kinect camera $\mathbb{C}_A$. The coarse-to-fine calibration of these 13 cameras is based on the detection of the reliable common point pairs across the cameras. The reliable point pair detection, coarse estimation and estimation refinement steps of the RGB-Kinect calibration are described in detail in the next three subsections.

3.3.2 Reliable point pair detection

Interest point detection plays a fundamental role in a lot of 3D vision-based applications [GHT11]. The traditional Scale-Invariant Feature Transform (SIFT) keypoint detector and descriptor [Low04b] are leveraged to detect feature points in the camera image spaces of $\mathbb{C}_A$ and $\mathbb{C}_j$ ($1 \leqslant j \leqslant 12$), which can then be used to construct reliable matched point pairs by taking advantage of the k-Nearest-Neighbors (KNN) [Alt92] and ratio test [Low04b] approaches. The resulting corresponding pairs still contain some outliers, which are filtered by using epipolar constraints [HZ03] with the RANdom SAmple Consensus (RANSAC) algorithm [FB81]. The remaining corresponding point pairs are assumed to be accurate for the next coarse estimation step.

3.3.3 Coarse estimation

Let n denote the number of the corresponding point pairs calculated in the previous subsection. Each corresponding point pair is represented by $(\boldsymbol{u}_i^j, \boldsymbol{u}_i^a)$, where $\boldsymbol{u}_i^j = \begin{pmatrix} u_i^j, & v_i^j, & 1 \end{pmatrix}^{\mathrm{T}}$ is a 2D point ($1 \leqslant i \leqslant n$) in the camera image space of $\mathbb{C}_j$ ($1 \leqslant j \leqslant 12$), and $\boldsymbol{u}_i^a$ denotes the corresponding 2D point in the camera image space of $\mathbb{C}_A$, i. e., $\boldsymbol{u}_i^a = \begin{pmatrix} u_i^a, & v_i^a, & 1 \end{pmatrix}^{\mathrm{T}}$. For each 2D point $\boldsymbol{u}_i^a$, the corresponding 3D point $\boldsymbol{x}_i^a = \begin{pmatrix} x_i^a, & y_i^a, & z_i^a, & 1 \end{pmatrix}^{\mathrm{T}}$ in the camera 3D space of $\mathbb{C}_A$ can be calculated by using the intrinsic camera matrix $\mathbf{K}_A$ and the depth information of $\mathbb{C}_A$ (refer to Equation 3.4.1). The coarse estimation describes the estimation of the rigid transformation $(\mathbf{R}_j^1, \boldsymbol{t}_j^1)$ from the camera coordinates of $\mathbb{C}_A$ to the camera coordinates of $\mathbb{C}_j$ ($1 \leqslant j \leqslant 12$) by solving the following optimization problem:

$$\min_{\mathbf{R}_j^1,\, \boldsymbol{t}_j^1} \sum_{i=1}^{n} \iota_{ij} \left\| \boldsymbol{u}_i^j - \hat{\boldsymbol{u}}\left(\boldsymbol{x}_i^a,\ \mathbf{K}_j,\ \mathbf{R}_j^1,\ \boldsymbol{t}_j^1\right) \right\|^2, \tag{3.3.1}$$

where

$$\hat{\boldsymbol{u}}(\boldsymbol{x},\ \mathbf{K},\ \mathbf{R},\ \boldsymbol{t}) = \mathbf{K}\,\mathrm{proj}\left(\left[\mathbf{R} \mid \boldsymbol{t}\right]\boldsymbol{x}\right). \tag{3.3.2}$$

Note that $\iota_{ij} \in \{0, 1\}$ denotes the visibility between the point $\boldsymbol{x}_i^a$ and the camera $\mathbb{C}_j$. The results of the coarse estimation stage are denoted by $(\mathbf{R}_j^1, \boldsymbol{t}_j^1)$, $1 \leqslant j \leqslant 12$, and the symbol '1' in the top right corner stands

for the coarse estimation stage. Several methods can be used for solving the PnP problem shown in Equation 3.3.1, e. g., LHM [LHM00], EPnP [LMF09] and RPnP [LXX12]. Based on the special camera arrangement of our 1D large-scale light field acquisition system, a camera orientation approximation-based PnP solution [GEK+17a] is proposed as below.

When observing the camera configuration in Figure 3.1, we find that $\mathbb{C}_A$ and $\mathbb{C}_j$ ($1 \leqslant j \leqslant 12$) have a minor orientation difference, implying that the rotation matrix $\mathbf{R}_j^1$ can be approximated by means of a linear method proposed in [Low04a]. Let an inhomogeneous 3-vector $\boldsymbol{r}_j = \begin{pmatrix} \alpha_j, & \beta_j, & \gamma_j \end{pmatrix}^{\mathrm{T}}$ represent the camera orientation difference between $\mathbb{C}_A$ and $\mathbb{C}_j$. The approximated rotation matrix $\mathbf{R}_j^1$ can then be written as

$$\mathbf{R}_j^1 = \begin{bmatrix} 1 & -\gamma_j & \beta_j \\ \gamma_j & 1 & -\alpha_j \\ -\beta_j & \alpha_j & 1 \end{bmatrix}. \tag{3.3.3}$$

After transforming the 2D point $\boldsymbol{u}_i^j$ on the image plane into a 2D point $\boldsymbol{p}_i^j = \begin{pmatrix} p_i^j, & q_i^j, & 1 \end{pmatrix}^{\mathrm{T}}$ on the normalized image plane of $\mathbb{C}_j$ by means of $\mathbf{K}_j$, i. e., $\boldsymbol{p}_i^j = \mathbf{K}^{-1}\boldsymbol{u}_i^j$ (see Equation 3.2.9), the rigid transformation can be written as

$$\begin{bmatrix} \mathbf{R}_j^1 & | & \boldsymbol{t}_j^1 \end{bmatrix} \boldsymbol{x}_i^a = \varsigma \boldsymbol{p}_i^j. \tag{3.3.4}$$

The scaling factor ς can be derived from the combination of Equation 3.3.3 and Equation 3.3.4, i. e.,

$$\varsigma = \begin{pmatrix} -\beta_j, & \alpha_j, & 1, & \boldsymbol{t}_j^1(3) \end{pmatrix} \boldsymbol{x}_i^a. \tag{3.3.5}$$

Afterwards, equation Equation 3.3.4 can be written as

$$\mathbf{A} \begin{pmatrix} \boldsymbol{r}_j \\ \boldsymbol{t}_j^1 \end{pmatrix} = \begin{pmatrix} p_i^j z_i^a - x_i^a \\ q_i^j z_i^a - y_i^a \end{pmatrix}, \tag{3.3.6}$$

where

$$\mathbf{A} = \begin{bmatrix} -p_i^j y_i^a & (p_i^j x_i^a + z_i^a) & -y_i^a & 1 & 0 & -p_i^j \\ -(q_i^j y_i^a + z_i^a) & q_i^j x_i^a & x_i^a & 0 & 1 & -q_i^j \end{bmatrix}. \tag{3.3.7}$$

The linear least-squares problem presented in Equation 3.3.6 and Equa-

tion 3.3.7 can be solved by using the Singular Value Decomposition (SVD) algorithm to compute a pseudo-inverse or using normal equations, requiring at least three corresponding point pairs, i. e., $n \geqslant 3$. The approximated rotation matrix $\mathbf{R}_j^1$ is then converted to a standard rotation matrix by normalization.

3.3.4 Estimation refinement

Due to the depth precision and flying pixel problems of the ToF sensor of any Kinect V2 device [WS16; SLK15; LSK+10; KBK+10], the 3D point $\boldsymbol{x}_i^a$ generated from the 2D point $\boldsymbol{u}_i^a$ is not equal to the ground truth 3D point in the camera coordinate system of $\mathbb{C}_A$, which can be further refined by solving the minimization problem as below:

$$\begin{aligned}\min_{\mathbf{R}_j^2,\ \mathbf{R}_a,\ \boldsymbol{t}_j^2,\ \boldsymbol{t}_a,\ \boldsymbol{x}_i^a} \Bigg(\sum_{j=1}^{12} \sum_{i=1}^{n} \iota_{ij} \left\| \boldsymbol{u}_i^j - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^a,\ \mathbf{K}_j,\ \mathbf{R}_j^2 \mathbf{R}_j^1,\ \boldsymbol{t}_j^1 + \boldsymbol{t}_j^2) \right\|^2 + \\ \sum_{i=1}^{n} \left\| \boldsymbol{u}_i^a - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^a,\ \mathbf{K}_A,\ \mathbf{R}_a,\ \boldsymbol{t}_a) \right\|^2 \Bigg).\end{aligned} \tag{3.3.8}$$

Here, the initial values of rotation matrices $\mathbf{R}_j^2$ ($1 \leqslant j \leqslant 12$) and $\mathbf{R}_a$ are 3×3 identity matrices, and the initial values of translation vectors $\boldsymbol{t}_j^2$ ($1 \leqslant j \leqslant 12$) and $\boldsymbol{t}_a$ are zero vectors. Note that the symbol '2' in the top right corner stands for the estimation refinement stage. Besides, the input parameters $\mathbf{R}_j^1$ and $\boldsymbol{t}_j^1$ are the results of the previous coarse estimation step. The nonlinear least-squares optimization problem defined in Equation 3.3.8 can be solved efficiently by a robust BA approach [Zac14].

The final result of the rigid transformation from the camera coordinates of $\mathbb{C}_A$ to the camera coordinates of $\mathbb{C}_j$ is expressed by using the results of both coarse estimation and estimation refinement stages as follows:

$$\begin{aligned}\mathbf{R}_j &= \mathbf{R}_j^2 \mathbf{R}_j^1 (\mathbf{R}_a)^{\mathrm{T}}, \\ \boldsymbol{t}_j &= \boldsymbol{t}_j^1 + \boldsymbol{t}_j^2 - \mathbf{R}_j \boldsymbol{t}_a.\end{aligned} \tag{3.3.9}$$

The above formula shows the extrinsic parameters of all RGB cameras in group one in the coordinate system of $\mathbb{C}_A$. The camera calibration process for Kinect camera $\mathbb{C}_B$ and all RGB cameras in group two can be performed

in the same way. The estimated extrinsic parameters of RGB cameras in group one are not in the same coordinate frame as those in group two. To tackle this issue, we need to estimate the rigid transformation from the coordinate system of $\mathbb{C}_A$ to the coordinate system of $\mathbb{C}_B$, which will be elaborated in the following section.

3.4 Kinect registration

As introduced in Section 3.3, the two Kinect V2 cameras mounted on the movable 1D large-scale light field acquisition system are denoted by $\mathbb{C}_A$ and $\mathbb{C}_B$, respectively. Besides, their intrinsic parameters are extracted from the factory calibration. The registration of these two Kinect V2 cameras is essentially to estimate the rigid transformation between them. Specifically, the estimation of the rigid transformation from the coordinate frame of $\mathbb{C}_A$ to the coordinate frame of $\mathbb{C}_B$ can be performed in coarse-to-fine manner [GMK18; GEK+17b], i. e., $\mathbf{T} = \mathbf{T}_2\mathbf{T}_1$. Here, $\mathbf{T}_1$ and $\mathbf{T}_2$ stand for the rigid transformation results of coarse estimation and estimation refinement, respectively. The intrinsic parameters of $\mathbb{C}_A$ or $\mathbb{C}_B$ contain the focal lengths f_x^j, f_y^j and the principal point $\left(c_x^j,\ c_y^j\right)$, where $j \in \{a, b\}$. The lens distortion coefficients are utilized to eliminate distortions before saving any pair of registered color and depth images, denoted by $\mathcal{C}^j$ and $\mathcal{D}^j$, of which both are located on the camera image plane of the ToF sensor in a Kinect V2. More details about the coarse estimation and estimation refinement of Kinect registration are described in the next two subsections.

3.4.1 Coarse estimation

A 2D marker that can be simultaneously captured by a pair of Kinect V2 cameras ($\mathbb{C}_A$ and $\mathbb{C}_B$) facilitates the construction of reliable corresponding 2D point pairs on these cameras [KND15]. One of these corresponding 2D point pairs is denoted by $\left(\boldsymbol{u}_i^a,\ \boldsymbol{u}_i^b\right)$, where $\boldsymbol{u}_i^j = \begin{pmatrix} u_i^j, & v_i^j, & 1 \end{pmatrix}^{\mathrm{T}}$, $j \in \{a, b\}$. The depth value d_i^j associated with a 2D point $\boldsymbol{u}_i^j$ can be acquired from the depth image $\mathcal{D}^j$, i. e., $d_i^j = \mathcal{D}^j\left(u_i^j,\ v_i^j\right)$. Let $\kappa : \mathbb{P}^2 \times \mathbb{R} \rightarrow \mathbb{P}^3$ denote

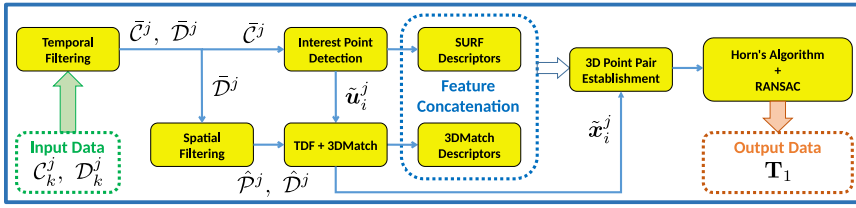


Figure 3.3. Flow chart of the proposed Kinect V2 registration method in the coarse estimation phase. (Source: [GMK18])

a back-projection function, which projects a 2D point $\boldsymbol{u}_i^j$ on the camera image plane to a 3D point $\boldsymbol{x}_i^j = \begin{pmatrix} x_i^j, & y_i^j, & z_i^j, & 1 \end{pmatrix}^{\mathrm{T}}$ in the camera space, i. e.,

$$\boldsymbol{x}_i^j = \kappa(\boldsymbol{u}_i^j,\ d_i^j) = \begin{pmatrix} \dfrac{(u_i^j - c_x^j)d_i^j}{f_x^j}, & \dfrac{(v_i^j - c_y^j)d_i^j}{f_y^j}, & d_i^j, & 1 \end{pmatrix}^{\mathrm{T}}. \tag{3.4.1}$$

Therefore, the 2D point pair $\left(\boldsymbol{u}_i^a,\ \boldsymbol{u}_i^b\right)$ can be turned into a 3D point pair $\left(\boldsymbol{x}_i^a,\ \boldsymbol{x}_i^b\right)$ by using Equation 3.4.1. The coarse rigid transformation $\mathbf{T}_1$ can then be estimated by solving

$$\arg \min_{\mathbf{R}_1 \in \mathrm{SO}(3),\, \boldsymbol{t}_1 \in \mathbb{R}^3} \sum_{i=1}^{n} \frac{1}{2} \left\| \begin{bmatrix} \mathbf{R}_1 & \boldsymbol{t}_1 \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a - \boldsymbol{x}_i^b \right\|_2^2. \tag{3.4.2}$$

The minimization problem in Equation 3.4.2 can be converted into the Orthogonal Procrustes problem [Sch66] and solved by the least-squares fitting algorithm [AHB87] efficiently, requiring at least three corresponding 3D point pairs, i. e., $n \geqslant 3$.

However, preparing some special calibration objects for the Kinect V2 registration task is sometimes time- and effort-consuming. How to solve the Kinect V2 registration problem by only leveraging the information from a real-world scene is more challenging than the above case of using a 2D marker. To this end, a novel coarse estimation framework [GMK18] is proposed and presented in Figure 3.3. This framework exploits both color and geometry feature descriptors to estimate a rough rigid transformation $\mathbf{T}_1$ between two Kinect V2 cameras. Specifically, this framework is

composed of eight steps as follows:

(i) **Input data preparation**. Due to the precision problem [WS16; SLK15; LSK+10; KBK+10] of the ToF sensor of any Kinect V2, multi-frame depth information is used to improve the quality of the captured depth images. For a static scene and a static multi-camera system, m consecutive depth and color frames are captured by both $\mathbb{C}_A$ and $\mathbb{C}_B$ simultaneously. The input data for the coarse estimation framework are $\mathcal{C}_k^j$ and $\mathcal{D}_k^j$, where $1 < k \leqslant m$ and $j \in \{a, b\}$.

(ii) **Temporal filtering**. A temporal mean filter is used here to calculate an average depth image $\bar{\mathcal{D}}^j$ from all the $\mathcal{D}_k^j$ images. Note that an underlying depth-validity check is also performed by this depth temporal mean filter. In particular, only depth image pixels with depth values larger than 0.5 m are regarded as valid pixels for the accumulated weights. A corresponding average color image $\bar{\mathcal{C}}^j$ is accordingly generated by using all the $\mathcal{C}_k^j$ images and the same accumulated weights with valid pixel positions from the depth temporal filtering process.

(iii) **Spatial filtering**. The mean depth image $\bar{\mathcal{D}}^j$ is then projected into a point cloud $\bar{\mathcal{P}}^j$ in the camera space of the Kinect V2 by using Equation 3.4.1. However, the resulting point cloud $\bar{\mathcal{P}}^j$ may still have some outliers or noisy data, of which some are far away from the real captured scene. This will increase the volume allocation for the volumetric representation described in the following steps, which may lead to a failure if limited memory is available in hardware, e. g., GPU. To handle this problem, a statistical spatial filtering method is utilized to trim the outliers of $\bar{\mathcal{P}}^j$. To be precise, each 3D point $\boldsymbol{x}_i^j$ in this point cloud has a mean distance t_i^j to its l nearest neighbor 3D points. A 3D point $\boldsymbol{x}_i^j$ will be removed if its distance t_i^j is not inside the range determined by the global distances mean and standard deviation. The filtered point cloud is denoted by $\hat{\mathcal{P}}^j$ and projected back onto the camera image plane by using a projection function

$\pi : \mathbb{P}^3 \to \mathbb{P}^2$, i. e.,

$$\pi(\boldsymbol{x}_i^j) = \left(\frac{f_x^j x_i^j}{z_i^j} + c_x^j, \quad \frac{f_y^j y_i^j}{z_i^j} + c_y^j, \quad 1 \right)^{\mathrm{T}}, \tag{3.4.3}$$

which generates a filtered depth image $\hat{\mathcal{D}}^j$ accordingly.

(iv) **Interest point detection**. The Speeded Up Robust Features (SURF) have robust and stable performance in computer vision and robotics applications [BTV06]. The SURF interest point detector is used to detect 2D keypoints on the average color image $\bar{\mathcal{C}}^j$ generated by the above temporal filtering step. The coordinates of all the keypoints are fed to the next step for geometry feature calculation. Besides, for each detected 2D interest point $\tilde{\boldsymbol{u}}_i^j$, the SURF algorithm also generates a SURF descriptor $\tilde{\boldsymbol{\omega}}_i^j \in \mathbb{R}^{64}$, which is a normalized vector.

(v) **TDF and 3DMatch**. The Truncated Distance Function (TDF) representation [CL96] is a variation of the Truncated Signed Distance Function (TSDF) representation [LC87]. The filtered point cloud $\hat{\mathcal{P}}^j$ is assigned to a volumetric grid of voxels to calculate the TDF value for each voxel. For each 2D interest point $\tilde{\boldsymbol{u}}_i^j$, a corresponding 3D interest point $\tilde{\boldsymbol{x}}_i^j$ can be computed by Equation 3.4.1 with its depth information from $\hat{\mathcal{D}}^j$. A volumetric 3D patch for each $\tilde{\boldsymbol{x}}_i^j$ is then extracted from the volumetric grid, i. e., $\tilde{\boldsymbol{x}}_i^j$ is in the center of a $30 \times 30 \times 30$ local voxel grid. Finally, the extracted volumetric 3D patch is fed to a pre-trained network of 3DMatch [ZSN+17] to generate a local geometry descriptor $\tilde{\boldsymbol{\epsilon}}_i^j \in \mathbb{R}^{512}$.

(vi) **Feature concatenation**. To make full use of different advantages of the SURF and 3DMatch descriptors for the scene representation, a feature concatenation strategy is proposed as below:

$$\tilde{\boldsymbol{\rho}}_i^j = \begin{pmatrix} (1-\lambda)\tilde{\boldsymbol{\omega}}_i^j \\ \lambda \tilde{\boldsymbol{\epsilon}}_i^j \end{pmatrix}, \tag{3.4.4}$$

where $\lambda \in [0,\ 1]$. The resulting concatenated descriptor is denoted by $\tilde{\boldsymbol{\rho}}_i^j \in \mathbb{R}^{576}$.

(vii) **3D point pair establishment**. After constructing the concatenated feature descriptor $\tilde{\rho}_i^j$ for each 3D interest point $\tilde{x}_i^j$, the reliable corresponding 3D point pairs in the two Kinect V2 camera spaces are established by means of the k-d tree data structure [Ben75] and KNN algorithm.

(viii) **Horn's algorithm and RANSAC**. The final rigid transformation $\mathbf{T}_1$ from $\mathbb{C}_A$ coordinates to $\mathbb{C}_B$ coordinates for the coarse estimation step is calculated by using the Horn's algorithm [Hor87] together with the RANSAC approach for solving the least squares problem presented in Equation 3.4.2.

3.4.2 Estimation refinement

The algorithm for the estimation refinement stage of the proposed Kinect registration method is depicted in Algorithm 1. It can be seen that the input data are the rigid transformation result $\mathbf{T}_1$ of the previous coarse estimation stage and point clouds $\hat{\mathcal{P}}^a$ and $\hat{\mathcal{P}}^b$ from the spatial filtering step (see step (iii) of Section 3.4.1). Moreover, it can also be seen that the estimation refinement is performed via three steps:

(i) **Transformation of coordinates**. The point cloud $\hat{\mathcal{P}}^a$ is transformed into the camera coordinate system of $\mathbb{C}_B$ by using the coarsely-estimated rigid transformation $\mathbf{T}_1$, so that the two point clouds are in the same camera space, i. e., $\mathbb{C}_B$ coordinates.

(ii) **Iterative point cloud registration**. The two point clouds in the same camera coordinate frame are registered by using an ICP-based method, which in this case is equal to the camera pose refinement.

(iii) **Estimation refinement result**. The final estimation refinement result $\mathbf{T}_2$ is recovered from two intermediate rigid transformation matrices $\mathbf{T}^a$ and $\mathbf{T}^b$, i. e., $\mathbf{T}_2 \leftarrow (\mathbf{T}^b)^{-1}\mathbf{T}^a$.

3.4.3 Others

The above coarse-to-fine Kinect registration enables us to describe the extrinsic parameters of all the 24 RGB cameras, which are estimated in

Algorithm 1: ICP-based estimation refinement algorithm.

Input : $\hat{\mathcal{P}}^j$ from step (iii) of Section 3.4.1, Rigid transformation $\mathbf{T}_1$.
Output: Rigid transformation $\mathbf{T}_2$.

/* Step 1: Transform $\hat{\mathcal{P}}^a$ from $\mathbb{C}_\mathrm{A}$ to $\mathbb{C}_\mathrm{B}$ coordinates */
foreach *point* $\boldsymbol{x}_i^a$ **in** $\hat{\mathcal{P}}^a$ **do** $\boldsymbol{x}_i^a \leftarrow \mathbf{T}_1 \boldsymbol{x}_i^a$;
/* Step 2: Iterative point cloud registration */
$\tau \leftarrow 0.005$;
$e \leftarrow +\infty$, $\check{e} \leftarrow 0$, $\dot{e} \leftarrow 0$;
$\mathbf{T}^a \leftarrow \mathbf{I}_4$, $\mathbf{T}^b \leftarrow \mathbf{I}_4$, $\dot{\mathbf{T}} \leftarrow \mathbf{I}_4$; /* $\mathbf{I}_4$: 4×4 identity matrix */
while *true* **do**
    $\check{\mathbf{T}}^a \leftarrow \mathbf{T}^a$;
    $\check{\mathbf{T}}^b \leftarrow \mathbf{T}^b$;
    $\check{e} \leftarrow e$;
    $e \leftarrow 0$;
    $\dot{\mathbf{T}}$, $\dot{e} \leftarrow$ ICP($\hat{\mathcal{P}}^a$, $\hat{\mathcal{P}}^b$); /* $\dot{e}$: Average error per point */
    foreach *point* $\boldsymbol{x}_i^a$ **in** $\hat{\mathcal{P}}^a$ **do** $\boldsymbol{x}_i^a \leftarrow \dot{\mathbf{T}} \boldsymbol{x}_i^a$;
    $\mathbf{T}^a \leftarrow \dot{\mathbf{T}} \mathbf{T}^a$;
    $e \leftarrow e + \dot{e}$;
    $\dot{\mathbf{T}}$, $\dot{e} \leftarrow$ ICP($\hat{\mathcal{P}}^b$, $\hat{\mathcal{P}}^a$);
    foreach *point* $\boldsymbol{x}_i^b$ **in** $\hat{\mathcal{P}}^b$ **do** $\boldsymbol{x}_i^b \leftarrow \dot{\mathbf{T}} \boldsymbol{x}_i^b$;
    $\mathbf{T}^b \leftarrow \dot{\mathbf{T}} \mathbf{T}^b$;
    $e \leftarrow e + \dot{e}$;
    if $e > \check{e}$ **then**
        $\mathbf{T}^a \leftarrow \check{\mathbf{T}}^a$;
        $\mathbf{T}^b \leftarrow \check{\mathbf{T}}^b$;
        break;
    if $\frac{\check{e}-e}{e} < \tau$ **then** break;
/* Step 3: Estimation refinement result */
$\mathbf{T}_2 \leftarrow (\mathbf{T}^b)^{-1} \mathbf{T}^a$.

Section 3.3, using the same coordinate system. In order to turn the light field raw data captured by our 1D large-scale light field acquisition system into standard SSLFs defined in Chapter 4, we first need to estimate the

least orthogonal distance line, i. e., the baseline, from the centers of all the RGB cameras. After this, all the RGB camera views are rectified w.r.t. the estimated baseline via a linear rectification algorithm [FTV00] to ensure that the focal planes of all the rectified RGB cameras become approximately coplanar [KZP+13]. Moreover, how to ensure color consistency across all the RGB cameras has been well studied in [FL15; LDX11; IW05; Jos04].

Chapter 4

Light Field Reconstruction

DSLF is a discrete representation of the 4D approximation of the plenoptic function parameterized by two parallel planes (camera plane and image plane) [LH96; GGS+96], where multi-perspective camera views are arranged in such a way that *the disparity ranges between adjacent views are less than or equal to one pixel* [Vag20; VBG20; BSV+19]. The parallel camera and image planes are illustrated in Figure 4.1, where the "st" plane stands for the camera plane and "uv" plane represents the image plane.

In real-world environments, a DSLF is extremely difficult to capture by the state-of-the-art light field acquisition systems, such as micro-lens array [PW12; GL10; LG09; NLB+05], multi-camera array [TSL+19; FBD+19; API19; WOO+19; SBV+17; JMA06; WJV+05; ZC04; YEB+02] and coded mask [MWB+13; BAL+12; AN10; LLW+08; VRA+07], due to their hardware limitations. However, the SSLFs, where *the disparity ranges between neighboring views are greater than one pixel*, can generally be captured by these systems. As a result, for light field scenes in real-world environments, the desired unknown 4D DSLFs are typically reconstructed from these captured 4D SSLFs. In this chapter, we aim to study how to reconstruct an unknown 4D DSLF from a real-world 4D SSLF. To this end, we split an input 4D SSLF into a set of horizontal- or vertical-parallax 3D SSLFs; therefore, the 3D DSLF reconstruction approaches can be exploited to solve the 4D DSLF reconstruction problem. Regarding 3D DSLF reconstruction, we introduce two representations of 3D SSLF in Section 4.1. Based on these two representations, two categories of methods to solve the 3D DSLF reconstruction problem are briefly introduced in Section 4.2. The details of these two categories of solutions are then described in Section 4.3 and Section 4.4, respectively. Section 4.5 presents another category, containing solutions combining the advantages of methods from the previous two

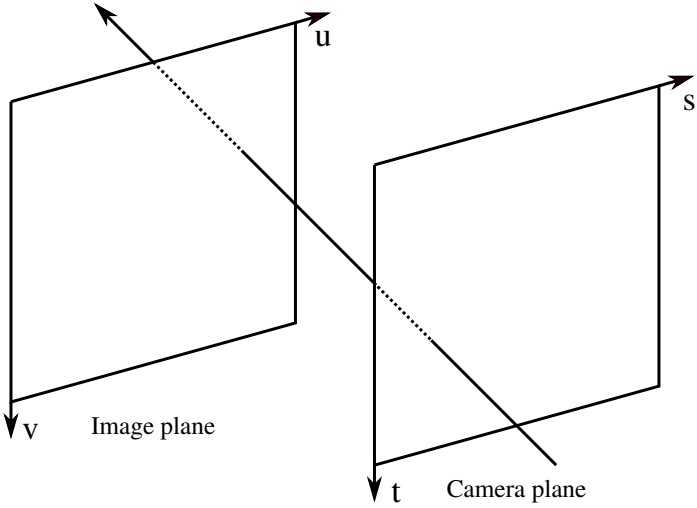
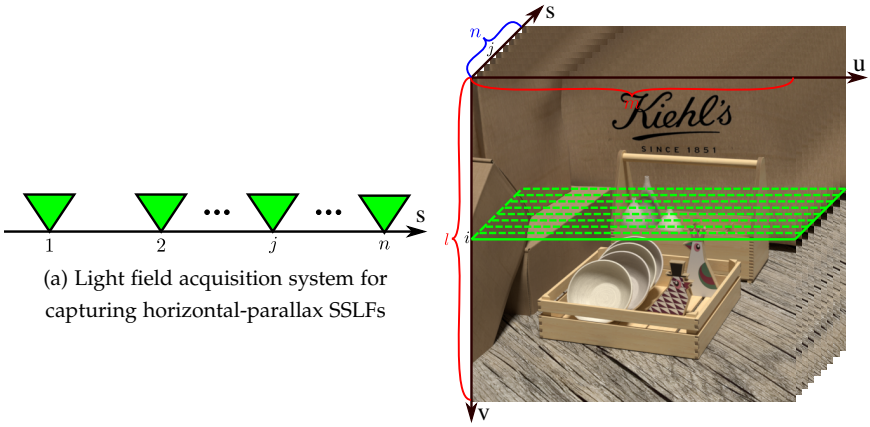


Figure 4.1. Image and camera planes of a 4D light field.

categories. Finally, two strategies of leveraging all the proposed 3D DSLF reconstruction approaches to solve the 4D DSLF reconstruction problem will be elaborated in Section 4.6.

4.1 3D SSLF representations

A horizontal-parallax 3D SSLF is a set of parallax images that are captured by a camera system shown in Figure 4.2 (a), where the cameras are evenly distributed along the axis 's' toward a common focus plane while keeping their optical axes parallel. Let a horizontal-parallax SSLF be represented by $\mathcal{S} = \{\mathcal{I}_j | 1 \leqslant j \leqslant n\}$, where n denotes the number of the parallax images in this 3D SSLF and $\mathcal{I}_j$ stands for one of these RGB parallax images and $\mathcal{I}_j \in \mathbb{R}^{m \times l \times 3}$. It is worth mentioning that n and $m \times l$ are also referred to as the angular and spatial resolutions of $\mathcal{S}$, respectively. After attaching all the parallax views of $\mathcal{S}$ along the axis 's', we construct a 3D light field volume as shown in Figure 4.2 (b). It is apparent that the 3D SSLF $\mathcal{S}$ can be treated as a set of sparsely-sampled EPIs, of which one is highlighted



(a) Light field acquisition system for capturing horizontal-parallax SSLFs

(b) 3D light field volume
shape = $m \times l \times n$ (u : v : s)
(Original image courtesy of [HJK+16])

Figure 4.2. 3D SSLF capture. A horizontal-parallax light field acquisition system in (a) captures a 3D SSLF, which is then turned into a 3D light field volume in (b).

with a green border. Consequently, the 3D SSLF $\mathcal{S}$ can also be written as $\mathcal{S} = \{\varepsilon_i | 1 \leqslant i \leqslant l\}$, where l is the number of the sparsely-sampled EPIs in $\mathcal{S}$ and all the sparsely-sampled RGB EPIs ε_i have the same size, i. e., $\varepsilon_i \in \mathbb{R}^{m \times n \times 3}$. To distinguish the two different ways when describing the 3D SSLF $\mathcal{S}$, the former one is referred to as parallax-view-based representation and the latter one is referred to as EPI-based representation.

The disparity information of the 3D SSLF $\mathcal{S}$ is important for the light field reconstruction methods presented in Section 4.4 and Section 4.5. Specifically, these methods require one to estimate the minimum disparity d_{min} and maximum disparity d_{max} of $\mathcal{S}$ [CSG+15]. These two disparity values can either be estimated by the state-of-the-art optical flow algorithms or by hands. In terms of the latter case, all the parallax images in $\mathcal{S}$ are sheared by several times using varying displacements with a measurement resolution of one pixel until the nearest and farthest focal planes corresponding to the nearest and farthest objects in the scene are found. The displacement values corresponding to these two planes are the

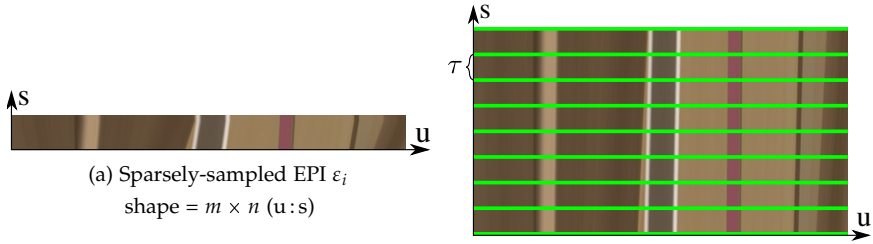
d_{min} and d_{max} of the 3D SSLF $\mathcal{S}$. The disparity range d_{range} of the 3D SSLF $\mathcal{S}$ can be derived accordingly, i. e., $d_{range} = (d_{max} - d_{min})$.

4.2 3D DSLF reconstruction

As introduced in the previous section about the two different representations of an input 3D SSLF $\mathcal{S}$, the target 3D DSLF $\mathcal{D}$ to be reconstructed from $\mathcal{S}$ can also be described by using these two representations. Specifically, $\mathcal{D}$ can be written either in the parallax-view-based form, i. e., $\mathcal{D} = \{\tilde{\mathcal{I}}_k | 1 \leqslant k \leqslant \dot{n}\}$, or in the EPI-based form, i. e., $\mathcal{D} = \{\zeta_i | 1 \leqslant i \leqslant l\}$. Here, $\dot{n}$ denotes the number of parallax images in the target 3D DSLF $\mathcal{D}$, i. e., the angular resolution of $\mathcal{D}$; $\tilde{\mathcal{I}}_k \in \mathbb{R}^{m \times l \times 3}$ stands for one of the parallax images in $\mathcal{D}$; and $\zeta_i \in \mathbb{R}^{m \times \dot{n} \times 3}$ represents one of the densely-sampled EPIs of $\mathcal{D}$. The relationship between the input 3D SSLF $\mathcal{S}$ and target 3D DSLF $\mathcal{D}$ is determined by the sampling interval τ, i. e., $\tau = \frac{\dot{n}-1}{n-1}$. There are two constraints for the sampling interval τ: (i) $\tau \in \mathbb{N}^*$ and (ii) $\tau \geqslant d_{range}$, because the disparity range of the target 3D DSLF $\mathcal{D}$ should be less than or equal to one pixel, i. e., $\frac{d_{range}}{\tau} \leqslant 1$. The proposed methods for solving the problem of reconstructing the target 3D DSLF $\mathcal{D}$ from the input 3D SSLF $\mathcal{S}$ can be categorized into two groups, as outlined in the following.

(i) **Novel view synthesis**. Typically, $\mathcal{D}$ contains n parallax views originally from $\mathcal{S}$, i. e., $\tilde{\mathcal{I}}_{\tau(j-1)+1} = \mathcal{I}_j$ $(1 \leqslant j \leqslant n)$. The remaining $(\dot{n} - n)$ parallax views in $\mathcal{D}$ can be generated from these n parallax views using novel view synthesis methods. Specifically, novel view synthesis methods synthesize $\frac{\dot{n}-n}{n-1} = (\tau - 1)$ novel parallax views between any two neighboring parallax views in $\mathcal{S}$.

(ii) **EPI inpainting**. Both $\mathcal{D}$ and $\mathcal{S}$ have the same number of EPIs, i. e., l. Each densely-sampled EPI ζ_i in $\mathcal{D}$ corresponds to a sparsely-sampled EPI ε_i in $\mathcal{S}$, where $1 \leqslant i \leqslant l$. However, the image resolution of ε_i is different from that of ζ_i. Specifically, the vertical resolution of ε_i is n, which is smaller than that of ζ_i, i. e., $\dot{n} = \big((n-1)\tau + 1\big)$. It means that performing 3D DSLF reconstruction on $\mathcal{S}$ is also equal to performing image inpainting or super-resolution on each ε_i of $\mathcal{S}$. Figure 4.3 (a) illustrates a sparsely-sampled EPI ε_i picked from the 3D light field



(a) Sparsely-sampled EPI ε_i
shape = $m \times n$ (u : s)

(b) Reconstructed densely-sampled EPI ζ_i
shape = $m \times \dot{n}$ (u : s)

Figure 4.3. EPI reconstruction. A densely-sampled EPI ζ_i in (b) is reconstructed from a sparsely-sampled EPI ε_i in (a), which is picked from the 3D light field volume in Figure 4.2 (b).

volume in Figure 4.2 (b). The corresponding reconstructed densely-sampled EPI ζ_i is illustrated in Figure 4.3 (b), where the green lines stand for the color rows from ε_i. It can be seen that there are $(\tau - 1)$ inpainted rows between any two neighboring green rows in ζ_i.

More details about the above two categories of solutions to the 3D DSLF reconstruction problem will be presented in Section 4.3 and Section 4.4, respectively.

4.3 Novel view synthesis

Since all the cameras in the light field acquisition system in Figure 4.2 (a) have the same camera specification and setup, the input 3D SSLF $\mathcal{S} = \{\mathcal{I}_j | 1 \leqslant j \leqslant n\}$ can be considered as a video captured by a virtual camera moving horizontally. The unknown $(\dot{n} - n)$ parallax views of $\mathcal{D} \backslash \mathcal{S}$ correspond to the unknown $(\dot{n} - n)$ intermediate frames of this video. Therefore, the 3D DSLF reconstruction problem can be solved by using video frame interpolation-based methods. In the following four subsections, we will first introduce two novel view synthesis approaches based on the video frame interpolation technique, i. e., SepConv and PIASC. Then we will show how to use them to address the 3D DSLF reconstruction problem in a recursive fashion. Finally, we will present how to perform novel view

synthesis by using the optical flow technique.

4.3.1 Separable Convolution (SepConv)

SepConv is one of the state-of-the-art video frame interpolation methods using deep learning techniques [NML17b]. Specifically, SepConv interpolates an intermediate frame between two input consecutive video frames by means of a CNN. Let the first two parallax views in $\mathcal{S}$, i. e., $\mathcal{I}_1$ and $\mathcal{I}_2$, be the input two frames and $\tilde{\mathcal{I}}$ be the target intermediate frame to be reconstructed. Essentially, the CNN of SepConv is aimed at generating two 2D convolution kernels, $\mathbf{K}_1(u, v)$ and $\mathbf{K}_2(u, v)$, for each 2D point with coordinates (u, v) on $\tilde{\mathcal{I}}$. A 2D kernel $\mathbf{K}_\mu(u, v)$, $\mu \in \{1, 2\}$, can be derived from a pair of separable 1D vectors $(\boldsymbol{V}^\mu_{u,v}, \boldsymbol{H}^\mu_{u,v})$, i. e.,

$$\mathbf{K}_\mu(u, v) = \boldsymbol{V}^\mu_{u,v}(\boldsymbol{H}^\mu_{u,v})^{\mathrm{T}}. \tag{4.3.1}$$

Here, $\boldsymbol{V}^\mu_{u,v} \in \mathbb{R}^\varrho$, $\boldsymbol{H}^\mu_{u,v} \in \mathbb{R}^\varrho$ and ϱ denotes the size of the separable 1D vectors. The color information for each 2D point on $\tilde{\mathcal{I}}$ is predicted by

$$\tilde{\mathcal{I}}(u, v, c) = \sum_{\mu=1}^{2} \big(\mathbf{K}_\mu(u, v) * \mathbf{P}_\mu(u, v, c)\big). \tag{4.3.2}$$

Here, the symbol '$*$' denotes the convolution operation and symbol 'c' stands for the color channel, i. e., $c \in \{r, g, b\}$. Besides, $\mathbf{P}_\mu(u, v, c)$ represents the image patch centered at (u, v) in the c channel of $\mathcal{I}_\mu$ ($\mu \in \{1, 2\}$). It has the same size as $\mathbf{K}_\mu(u, v)$, i. e., $\varrho \times \varrho$. Compared to the adaptive convolution method proposed in [NML17a], the SepConv approach reduces the number of unknown kernel parameters from $2m \times l \times \varrho^2$ to $2m \times l \times 2\varrho$, thereby enabling a high-resolution synthesized view to be generated at once efficiently. Moreover, the amount of object motion in a video that can be handled by SepConv is restricted to ϱ ($= 51$) as explained in [NML17b].

4.3.2 Parallax-Interpolation Adaptive Separable Convolution (PIASC)

The above SepConv method is originally designed for novel frame synthesis for videos containing objects moving in different directions at varying

speeds. However, the horizontal-parallax input 3D SSLF $\mathcal{S}$ is treated as a video captured by a horizontally moving virtual camera here, which means that the parallax views in $\mathcal{S}$ contain objects moving in one direction, i. e., left or right, at a fixed speed. Additionally, constructing a dedicated deep fully CNN based on SepConv for the purpose of 3D DSLF reconstruction is not always easy, considering that public high-resolution and high-quality real-world light field datasets are not as common as public high-definition and high-fidelity real-world videos and the training process would be enormously time- and effort-consuming. In order to take full advantage of the state-of-the-art video frame interpolation method, SepConv, for tackling the 3D DSLF reconstruction problem and to avoid its cumbersome re-training process, a novel 3D DSLF reconstruction method, PIASC, is proposed in [GK18]. Specifically, PIASC is a fine-tuning strategy that adapts the motion-sensitive convolution kernels of SepConv to perform video frame interpolation for videos having objects moving in one direction at a uniform speed. All the coefficient values in the convolution kernel in Equation 4.3.1 are adjusted by this fine-tuning strategy via an elaborately-designed weight matrix $\mathbf{W}$ as below:

$$\hat{\mathbf{K}}_{\mu}(u, v) = \frac{\varrho^2}{\sum_{x=1}^{\varrho}\sum_{y=1}^{\varrho} \mathbf{W}(x, y; \sigma)} \mathbf{W} \odot \mathbf{K}_{\mu}(u, v), \tag{4.3.3}$$

where

$$\mathbf{W}(x, y; \sigma) = \exp\left(-\frac{1}{2}\left(\frac{|y - \bar{\varrho}|}{\sigma} \right)^2 \right), \tag{4.3.4}$$

$$x, y \in [1, \varrho] \text{ and } \bar{\varrho} = \frac{\varrho + 1}{2}. \tag{4.3.5}$$

Here, the symbol '$\odot$' denotes the element-wise (Hadamard) product and $\hat{\mathbf{K}}_{\mu}(u, v)$ represents the horizontal-motion-enhanced convolution kernel adapted from SepConv. The weight matrix $\mathbf{W}$ is similar to a Gaussian kernel, of which the shape is determined by σ (= 200 here); however, only the coordinate information along the vertical axis of $\mathbf{W}$ is taken into account by PIASC, which is because of the aforementioned feature that objects in different parallax images of the input 3D SSLF $\mathcal{S}$ only have

Algorithm 2: Recursive interpolation algorithm for reconstructing a 3D DSLF $\mathcal{D}$ from a 3D SSLF $\mathcal{S}$ using PIASC or SepConv.

Input: 3D SSLF $\mathcal{S} = \{\mathcal{I}_j | 1 \leqslant j \leqslant n\}$;
Sampling interval $\tau \in \{2, 4, 8, 16, 32, \ldots\}$.
Output: 3D DSLF $\mathcal{D} = \{\tilde{\mathcal{I}}_k | 1 \leqslant k \leqslant \dot{n}\}$, where $\dot{n} = \big((n-1)\tau + 1\big)$.

/* range(1, $\dot{n}$, τ)= $\{1, 1+\tau, 1+2\tau, \ldots, \dot{n}\}$ */
1 **for** ω **in** range(1, $\dot{n}$, τ) **do**
2 $\tilde{\mathcal{I}}_\omega \leftarrow \mathcal{I}_{\left(\frac{\omega-1}{\tau}+1\right)}$;
3 **while** $\tau > 1$ **do**
4 $\dot{\tau} \leftarrow \frac{\tau}{2}$;
5 **for** ω **in** range(1, $\dot{n} - \tau$, τ) **do**
6 $\tilde{\mathcal{I}}_{\omega+\dot{\tau}} \leftarrow$ PIASC($\tilde{\mathcal{I}}_\omega$, $\tilde{\mathcal{I}}_{\omega+\tau}$);
  /* or $\tilde{\mathcal{I}}_{\omega+\dot{\tau}} \leftarrow$ SepConv($\tilde{\mathcal{I}}_\omega$, $\tilde{\mathcal{I}}_{\omega+\tau}$); */
7 $\tau \leftarrow \dot{\tau}$;

horizontal displacements.

4.3.3 Recursive interpolation

The SepConv and PIASC approaches introduced in the previous two subsections are designed for synthesizing only one intermediate view between any two adjacent parallax views in the input 3D SSLF $\mathcal{S}$. However, depending on the different sampling interval τ (see Section 4.2), the number of the desired intermediate views between any two neighboring parallax views in $\mathcal{S}$, i. e., $(\tau - 1)$, may be greater than one. To generate more than one novel view between two input parallax views, the above two video frame interpolation-based novel view synthesis solutions can be applied in a recursive manner. The overall process of the recursive interpolation algorithm for 3D DSLF reconstruction on a 3D SSLF $\mathcal{S}$ using PIASC or SepConv is depicted in Algorithm 2. All the parallax images in $\mathcal{S}$ are first used to recover some of the missing images of the target unknown 3D DSLF $\mathcal{D}$ as shown in lines 1-2. The unknown view, i. e., $\tilde{\mathcal{I}}_{\omega+\dot{\tau}}$,

in the middle of adjacent reconstructed views, i. e., $\tilde{\mathcal{I}}_{\omega}$ and $\tilde{\mathcal{I}}_{\omega+\tau}$, is then synthesized by using the proposed PIASC or SepConv method, which is shown in line 6. Finally, this operation is repeated recursively until all the $(\dot{n}-n)$ unknown parallax views in $\mathcal{D}\backslash\mathcal{S}$ are reconstructed. The only requirement of the recursive interpolation algorithm is that the sampling interval τ should be a power of two, i. e., $\log_2\tau = \lceil \log_2\tau \rceil$. It implies that this recursive interpolation algorithm is not a universal solution to the 3D DSLF reconstruction problem when given an arbitrary τ.

4.3.4 Optical flow

As discussed above, the recursive interpolation algorithm based on PIASC or SepConv can hardly solve the problem of reconstructing a 3D DSLF $\mathcal{D}$ from an input 3D SSLF $\mathcal{S}$ with an arbitrary sampling interval τ. The optical flow technique is more flexible than the recursive interpolation algorithm for synthesizing an intermediate frame at an arbitrary time step between two input video frames and, therefore, can also be used for solving the 3D DSLF reconstruction problem. Specifically, optical flow is capable of recovering an intermediate parallax view at any arbitrary space step between two neighboring parallax views in the input 3D SSLF $\mathcal{S}$. Let t denote this arbitrary space step, where $t \in \{\frac{1}{\tau}, \frac{2}{\tau}, \ldots, \frac{\tau-1}{\tau}\}$. The $(\dot{n}-n)$ missing parallax views in $\mathcal{D}\backslash\mathcal{S}$ can be reconstructed as below:

$$\mathcal{I}_{j+t} = \Lambda \odot g\Big(\mathcal{I}_j,\ \mathcal{F}_{(j+t)\to j}\Big) + (1-\Lambda)\odot g\Big(\mathcal{I}_{j+1},\ \mathcal{F}_{(j+t)\to(j+1)}\Big), \quad (4.3.6)$$

$$\Lambda = \frac{(1-t)\mathcal{V}_{(j+t)\leftarrow j}}{(1-t)\mathcal{V}_{(j+t)\leftarrow j} + t\Big(1-\mathcal{V}_{(j+t)\leftarrow j}\Big)}. \quad (4.3.7)$$

Here, the target intermediate image $\mathcal{I}_{j+t}$ denotes the image $\tilde{\mathcal{I}}_{(j-1+t)\tau+1}$ of $\mathcal{D}$; $\mathcal{F}_{(j+t)\to j}$ and $\mathcal{F}_{(j+t)\to(j+1)}$ represent the optical flow maps from $\mathcal{I}_{j+t}$ to $\mathcal{I}_j$ and $\mathcal{I}_{j+t}$ to $\mathcal{I}_{j+1}$, respectively; $g(\cdot,\cdot)$ stands for the inverse warping function using bilinear interpolation [JSJ+18]; and $\mathcal{V}_{(j+t)\leftarrow j}$ is a soft visibility map from $\mathcal{I}_j$ to $\mathcal{I}_{j+t}$ with the same size as them. However, the inverse optical flow maps, i. e., $\mathcal{F}_{(j+t)\to j}$ and $\mathcal{F}_{(j+t)\to(j+1)}$ in Equation 4.3.6, can hardly be computed, mainly because the target parallax view $\mathcal{I}_{j+t}$ is unknown. However, the bidirectional optical flow maps, i. e., $\mathcal{F}_{j\to(j+1)}$ and

$\mathcal{F}_{(j+1)\to j}$, between $\mathcal{I}_j$ and $\mathcal{I}_{j+1}$, can easily be estimated by different state-of-the-art optical flow approaches [XCW+19; SYL+18; MHR18; IMS+17]. Therefore, the inverse optical flows are typically approximated from the bidirectional optical flow as below:

$$\begin{aligned}\tilde{\mathcal{F}}_{(j+t)\to j} &= -(1-t)t\mathcal{F}_{j\to(j+1)} + t^2\mathcal{F}_{(j+1)\to j},\\ \tilde{\mathcal{F}}_{(j+t)\to(j+1)} &= (1-t)^2\mathcal{F}_{j\to(j+1)} - t(1-t)\mathcal{F}_{(j+1)\to j}.\end{aligned} \tag{4.3.8}$$

How to estimate the bidirectional optical flow maps $\mathcal{F}_{j\to(j+1)}$ and $\mathcal{F}_{(j+1)\to j}$ in Equation 4.3.8 and soft visibility map $\mathcal{V}_{(j+t)\leftarrow j}$ in Equation 4.3.7 will be elaborated in Section 4.5.2.

4.4 EPI inpainting

In the previous section, the novel view synthesis-based 3D DSLF reconstruction considers the input 3D SSLF $\mathcal{S}$ as a video; therefore, the target 3D DSLF $\mathcal{D}$ can be reconstructed from $\mathcal{S}$ by using video frame interpolation methods. In this section, we approach the 3D DSLF reconstruction problem from a different perspective. Specifically, the input 3D SSLF $\mathcal{S}$ is treated as a set of sparsely-sampled EPIs, i. e., $\mathcal{S} = \{\varepsilon_i | 1 \leqslant i \leqslant l\}$. Image inpainting techniques are exploited to reconstruct the densely-sampled EPI ζ_i in $\mathcal{D}$ from the sparsely-sample EPI ε_i in $\mathcal{S}$. After inpainting all the sparsely-sampled EPIs in $\mathcal{S}$, the target 3D DSLF $\mathcal{D} = \{\zeta_i | 1 \leqslant i \leqslant l\}$ can be reconstructed. In the following three subsections, we will present three different EPI inpainting methods, i. e., ST, MAST and DRST, respectively.

4.4.1 Shearlet Transform (ST)

ST is an IBR technique that is based on shearlet transform [KLR16; KSZ12; KL12] and designed for 3D DSLF reconstruction from 3D SSLFs by means of the sparsity of EPIs in shearlet domain [VBG18; VBG17; VBG15]. Specifically, this 3D DSLF reconstruction approach (ST) consists of four steps: (i) pre-shearing, (ii) shearlet system construction, (iii) sparse regularization and (iv) post-shearing. Steps (i), (ii) and (iv) rely on the disparity estimation of the input 3D SSLF as introduced in Section 4.1. In particular, the estimated minimum disparity d_{min}, maximum disparity d_{max} and disparity

range d_{range} of the input 3D SSLF are exploited to rearrange the rows of each sparsely-sampled EPI via shearing and zero padding operations and to construct a suitable specifically-tailored universal shearlet system. The sparse regularization step is the core of ST, which is used for removing the aliasing artifacts of EPIs in frequency domain. It is composed of shearlet analysis transform, hard thresholding, shearlet synthesis transform and an optional double overrelaxation (DORE) algorithm [VBG17], which is capable of accelerating the convergence speed of the whole sparse regularization step effectively. More details about the four steps of ST are described as follows.

(i) **Pre-shearing**. Since the shearlet filters of the specifically-tailored shearlet system are associated with varying disparities of EPIs in the range of $0 - \tau$ pixels, the input SSLF $\mathcal{S}$ may require to be sheared in order to make full use of all the filters. The pre-shearing operation with parameter φ is used to uniformly shear all the parallax images of the input 3D SSLF $\mathcal{S}$, such that the horizontal displacement between neighboring sheared images is equal to φ pixels. An example is given in Figure 4.6, where (b) shows the result of applying the pre-shearing operation to the input sparsely-sampeld EPI in (a). In addition, the shearing parameter φ is decided by the sampling interval τ, minimum disparity d_{min} and disparity range d_{range} of $\mathcal{S}$, i. e., $\big(d_{min} - \big(\tau - d_{range}\big)\big) \leqslant \varphi \leqslant d_{min}$. More details can also be found in Section 4.4.3 (ii).

(ii) **Shearlet system construction**. For scaling function $\phi \in L^2(\mathbb{R}^2)$, two shearlets $\psi, \tilde{\psi} \in L^2(\mathbb{R}^2)$ and parameter $c = (c_1, c_2) \in (\mathbb{R}_+)^2$, a regular cone-adapted discrete 2D shearlet system $\mathcal{SH}(\phi, \psi, \tilde{\psi}; c)$ is defined by

$$\mathcal{SH}(\phi, \psi, \tilde{\psi}; c) = \begin{cases} \phi_m = \phi(\cdot - c_1 m) \\ \psi_{j,k,m} = 2^{\frac{j+\lfloor j/2 \rfloor}{2}} j \psi(S_k A_{2^j} \cdot - M_c m) \\ \tilde{\psi}_{j,k,m} = 2^{\frac{j+\lfloor j/2 \rfloor}{2}} j \tilde{\psi}(S_k^T \tilde{A}_{2^j} \cdot - \tilde{M}_c m) \end{cases} \tag{4.4.1}$$

where

$$A_{2^j} = \begin{bmatrix} 2^j & 0 \\ 0 & 2^{j/2} \end{bmatrix}, \ \tilde{A}_{2^j} = \begin{bmatrix} 2^{j/2} & 0 \\ 0 & 2^j \end{bmatrix} \tag{4.4.2}$$

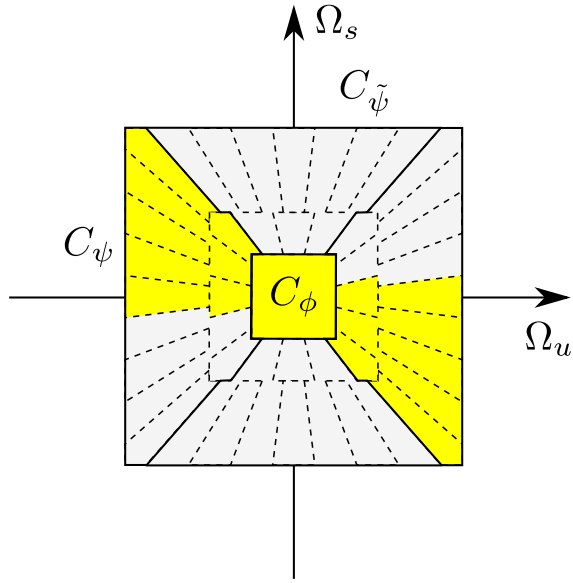


Figure 4.4. Frequency plane tiling by the shearlet transform using the regular cone-adapted discrete 2D shearlet system with $\xi = 2$ scales. The symbol 'C_ϕ' denotes the low-frequency region. The symbols 'C_ψ' and '$C_{\tilde{\psi}}$' represent the horizontal and vertical conic regions, respectively. The yellow partitions correspond to the specifically-tailored shearlet system proposed in [VBG18]. (Source: [VBG18])

are parabolic scaling matrices;

$$S_k = \begin{bmatrix} 1 & k \\ 0 & 1 \end{bmatrix} \tag{4.4.3}$$

is a shearing matrix; and

$$M_c = \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix}, \tilde{M}_c = \begin{bmatrix} c_2 & 0 \\ 0 & c_1 \end{bmatrix} \tag{4.4.4}$$

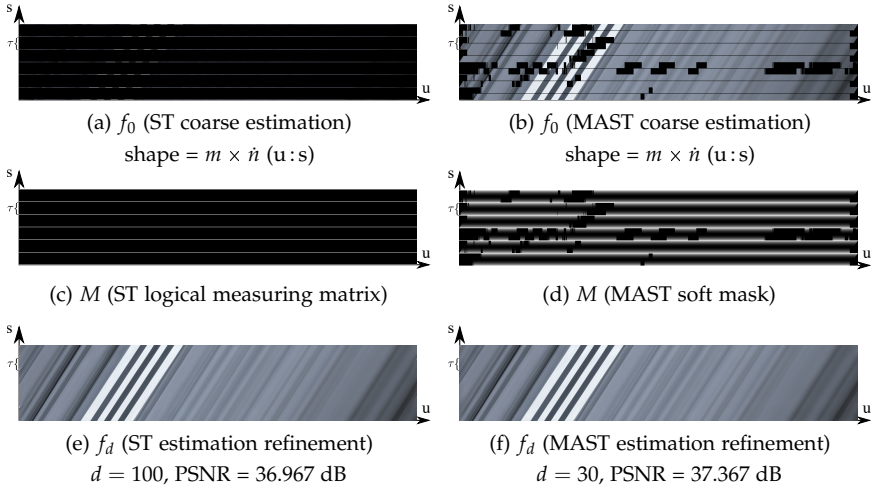
are sampling densities of the 2D translation grid for a spatial position $m \in \mathbb{Z}^2$. For each scale $j \in \mathbb{N}^*$, the shearing parameter $|k| \leqslant \frac{2^j+1}{2}$ and $k \in \mathbb{Z}$. In addition, ξ denotes the number of the scales of the universal shearlet system; therefore, i. e., $1 \leqslant j \leqslant \xi$. In [VBG18], a

specifically-tailored shearlet system is extracted from the universal shearlet system to solve the EPI reconstruction problem. As shown in Figure 4.4, the specifically-tailored shearlet system is represented by the yellow partitions in frequency domain, which correspond to the disparity values, i. e., $0 - \tau$ pixels, of the input EPIs in image domain. The number of the filters of the specifically-tailored shearlet system, i. e., η, is determined by the number of the scales, i. e., ξ, with an equation $\eta = (2^{\xi+1} + \xi - 1)$. Based on the specifically-tailored shearlet system, the shearlet analysis transform is denoted by $\mathcal{SH} : \mathbb{R}^{\gamma \times \gamma} \rightarrow \mathbb{R}^{\gamma \times \gamma \times \eta}$ and shearlet synthesis transform is written as $\mathcal{SH}^* : \mathbb{R}^{\gamma \times \gamma \times \eta} \rightarrow \mathbb{R}^{\gamma \times \gamma}$, where $\gamma \times \gamma$ represents the spatial resolution of each shearlet filter.

(iii) **Sparse regularization**. Given a sparsely-sampled EPI $\varepsilon \in \mathbb{R}^{m \times n}$ with disparities in the range of $0 - \tau$ pixels, the sparse regularization step recovers an unknown densely-sample EPI $\zeta \in \mathbb{R}^{m \times \dot{n}}$ by means of an iterative image inpainting algorithm. This algorithm exploits the sparse representation of ζ in shearlet domain. Note that the input coarsely-sampled EPI ε and target densely-sampled EPI ζ discussed here are grayscale images. As shown in Figure 4.5 (a), the sampling interval τ is leveraged to rearrange the rows of the input sparsely-sampled EPI ε. Specifically, each two neighboring rows in ε is padded with $(\tau - 1)$ black rows with values of zero. Let f_0 denote the zero-padded version of ε and M represent the logical measuring matrix associated with f_0, which is illustrated in Figure 4.5 (c), where the rows that are originally from ε have the value of one and the other rows have the value of zero. Besides, f_0 is also referred to as the coarse estimation of the target densely-sampled EPI ζ. Typically, the densely-sampled EPI ζ can be reconstructed from the input sparsely-sampled EPI f_0 by solving the following optimization problem in the shearlet transform domain:

$$\min_{\zeta} \ \|\mathcal{SH}(\zeta)\|_1 \,, \ \text{s.t.} \ f_0 = M \odot \zeta \,. \tag{4.4.5}$$

The sparse regularization of ST is successful in solving the above problem via an iterative inpainting process with d iterations, corresponding to the intermediate reconstructed EPI results f_{i+1}, where



(a) f_0 (ST coarse estimation) shape = $m \times \dot{n}$ (u : s)

(b) f_0 (MAST coarse estimation) shape = $m \times \dot{n}$ (u : s)

(c) M (ST logical measuring matrix)

(d) M (MAST soft mask)

(e) f_d (ST estimation refinement) $d = 100$, PSNR = 36.967 dB

(f) f_d (MAST estimation refinement) $d = 30$, PSNR = 37.367 dB

Figure 4.5. Coarse estimation, measuring matrix (soft mask) and estimation refinement of the target densely-sampled EPI ζ. (Source: [GBG+19])

$0 \leqslant i \leqslant d-1$. Particularly, for iteration i, the intermediate reconstruction result f_{i+1} is generated by using the DORE-based iterative algorithm proposed in [VBG17], which has been demonstrated to be faster and more robust than the original iterative hard thresholding algorithm presented in [VBG18]. Mathematically, this process can be written as below:

$$\begin{aligned} \hat{f}_i &= \mathcal{SH}^* \Big(T_{\lambda_i} \Big(\mathcal{SH}\big(f_i + \alpha(f_0 - M \odot f_i)\big) \Big) \Big) , \\ \tilde{f}_i &= \hat{f}_i + \beta_1 (\hat{f}_i - f_{i-1}) , \\ f_{i+1} &= \tilde{f}_i + \beta_2 (\tilde{f}_i - f_{i-2}) , \end{aligned} \tag{4.4.6}$$

where

$$\begin{aligned} \beta_1 &= \frac{\operatorname{sum}\big((f_0 - \hat{f}_i) \odot M \odot (\hat{f}_i - f_{i-1})\big)}{\operatorname{sum}\big((\hat{f}_i - f_{i-1}) \odot M \odot (\hat{f}_i - f_{i-1})\big)} , \\ \beta_2 &= \frac{\operatorname{sum}\big((f_0 - \tilde{f}_i) \odot M \odot (\tilde{f}_i - f_{i-2})\big)}{\operatorname{sum}\big((\tilde{f}_i - f_{i-2}) \odot M \odot (\tilde{f}_i - f_{i-2})\big)} . \end{aligned} \tag{4.4.7}$$

Here, $\mathrm{sum}(\cdot)$ returns the sum of all the elements in the input matrix; α is a parameter for adjusting the convergence speed; $T_{\lambda_i}(\cdot)$ is a hard-thresholding operator [LLS+13] for the threshold level λ_i, which linearly decreases from λ_{max} to λ_{min} with iteration i increasing from 0 to $d-1$. Additionally, f_d is displayed in Figure 4.5 (e) and referred to as the estimation refinement of the target densely-sampled EPI ζ. As can be seen from Equation 4.4.6 and Equation 4.4.7, the computation time of the ST approach above is linearly dependent on the number of iterations, i. e., d. Therefore, a reliable f_0, i. e., the coarse estimation of ζ, would make it feasible to accelerate ST with a smaller d.

(iv) **Post-shearing**. To produce the final reconstructed target DSLF, we need to compensate for the shearing parameter φ, applied in the previous pre-shearing step, for all the reconstructed densely-sampled EPIs. More details can also be found in [GBG20].

Moreover, the TensorFlow implementation of the above four steps of ST is introduced in [GKB+19c].

4.4.2 Mask-Accelerated Shearlet Transform (MAST)

As discussed in the previous subsection, a reliable coarse estimation of the target densely-sampled EPI ζ facilitates the acceleration of the sparse regularization step of ST. To this end, a novel ST-based coarse-to-fine DSLF reconstruction method, MAST [GBG+19], is presented in this subsection. In order to make f_0 a better coarse estimation of the target densely-sampled EPI ζ, one of the state-of-the-art learning-based optical flow methods, i. e., FlowNet2 [IMS+17], is utilized to estimate the bidirectional optical flow between adjacent views in the input horizontal-parallax SSLF $\mathcal{S} = \{\mathcal{I}_j | 1 \leqslant j \leqslant n\}$. Since this SSLF contains no vertical object motions between any two neighboring views, the horizontal components of the estimated optical flow displacement vectors are treated as disparities. As defined in Section 4.3.4, the bidirectional optical flow maps between $\mathcal{I}_j$ and $\mathcal{I}_{j+1}$ in the input $\mathcal{S}$ are $F_{j\rightarrow(j+1)}$ and $F_{(j+1)\rightarrow j}$. A forward-backward consistency constraint [HR17] between $F_{j\rightarrow(j+1)}$ and $F_{(j+1)\rightarrow j}$ is utilized to roughly remove the inaccuracies caused by occlusions or large disparities

of objects. Let[1] $\dot{k} = (k-1)\%\tau$ and $j = 1 + (k - \dot{k} - 1)/\tau$, the estimated bidirectional optical flow is then used to perform a coarse estimation of the parallax images in $\mathcal{D} = \{\tilde{\mathcal{I}}_k | 1 \leqslant k \leqslant \dot{n}\}$ as follows[2]:

$$\tilde{\mathcal{I}}_k = \begin{cases} \mathcal{I}_j & \text{for} \quad \dot{k} = 0\,, \\ g(\mathcal{I}_j,\ -\frac{\dot{k}}{\tau} F_{j\to(j+1)}) & \text{for} \quad 0 < \dot{k} < \frac{\tau}{2}\,, \\ g(\mathcal{I}_{j+1},\ -\frac{(\tau-\dot{k})}{\tau} F_{(j+1)\to j}) & \text{for} \quad \frac{\tau}{2} < \dot{k} < \tau\,, \\ \mathbf{0} & \text{for} \quad \dot{k} = \frac{\tau}{2}\,. \end{cases} \tag{4.4.8}$$

Here, the inverse warping function $g(\cdot, \cdot)$ takes advantage of bicubic interpolation [Got03]. The roughly-estimated $\mathcal{D}$ is then turned into densely-sampled EPIs, such that the coarse estimation f_0 of ζ is partially restored as displayed in Figure 4.5 (b). Note that the large missing areas are caused by the filtering of the unreliable optical flow maps using the bidirectional consistency check. However, the roughly inpainted areas in f_0 are not accurate enough for directly using ST. Specifically, due to the accumulation error of the optical flow in the interpolation algorithm in Equation 4.4.8, horizontal lines of f_0 near the locations, i. e., $\dot{k} = \frac{\tau}{2}$, have larger inpainting errors than those near the ground-truth regions, i. e., $\dot{k} = 0$. The proposed MAST method solves this problem by replacing the logical measuring matrix in Equation 4.4.6 and Equation 4.4.7 with an elaborately-designed soft mask as below:

$$M(u, k) = \begin{cases} 1.0 & \text{for} \quad \dot{k} = 0\,, \\ \varpi(1 - \frac{2\dot{k}}{\tau})^2 & \text{for} \quad \dot{k} > 0\,,\ f_0(u, k) > 0\,, \\ 0 & \text{for} \quad \dot{k} > 0\,,\ f_0(u, k) = 0\,, \end{cases} \tag{4.4.9}$$

where $\varpi \in (0, 1)$, $1 \leqslant k \leqslant \dot{n}$ and $1 \leqslant u \leqslant m$. The new soft mask corresponding to f_0 is illustrated in Figure 4.5 (d). It can be seen that this soft mask suppresses the contributions of f_0 in the regions which are not inpainted or meet the condition that $\dot{k}$ is close to $\frac{\tau}{2}$; however, it enhances the contributions from the ground-truth nearby areas, thereby effectively improving the initialization of the densely-sampled EPIs for the iterative sparse regularization algorithm in Section 4.4.1 (iii). The densely-sampled EPI f_d reconstructed from f_0 by means of sparse regularization is illus-

[1] Here, the symbol '%' stands for the modulo operation.
[2] Assume that $\tau\%2 = 0$ for this $\mathcal{D}$.

trated in Figure 4.5 (f). From the figure we can see that the number of iterations used by the sparse regularization of MAST, i. e., $d = 30$, is much smaller than that of ST, i. e., $d = 100$ in Figure 4.5 (e).

4.4.3 Deep Residual Shearlet Transform (DRST)

The methods ST and MAST presented in the previous two subsections are based on the traditional model-based sparse regularization. In this subsection, a novel data-driven and learning-based DSLF reconstruction approach, DRST [GBK+20], is proposed by fully leveraging the state-of-the-art deep learning techniques and self-supervised learning [JT20]. Similar to ST, DRST also consists of four steps: (i) shearlet system reconstruction, (ii) pre-shearing and zero-padding, (iii) learning-based sparse regularization and (iv) post-shearing. It is worth mentioning that the proposed DRST is targeted at solving DSLF reconstruction problem for input SSLFs with moderate disparity ranges, i. e., $8 < d_{range} \leqslant 16$ pixels. More details of DRST are described as follows.

(i) **Shearlet system construction**. We fix the sampling interval τ to 16 for the input SSLFs with moderate disparity ranges, since this enables the constraint for τ, i. e., $\tau \geqslant d_{range}$, to be satisfied (see Section 4.2). Therefore, a corresponding shearlet system can be prepared, i. e., $\xi = \lceil \log_2 \tau \rceil = 4$ and $\eta = (2^{\xi+1} + \xi - 1) = 35$. The spatial resolution of the shearlet filters in this shearlet system is specified by the users. We use $\gamma = 127$ as recommended by [VBG18].

(ii) **Pre-shearing and zero-padding**. For better understanding the pre-shearing and zero-padding strategies in this subsection and how to leverage the synthetic SSLF data for training, we select the first-row horizontal-parallax light field of the 4D light field "Boxes" in the training dataset, i. e., the 4D light field dataset [HJK+16], as the input 3D SSLF $\mathcal{S}$ for demonstration. The input 3D light field $\mathcal{S}$ has an angular resolution of 9 and a spatial resolution of 512×512 pixels. Besides, the ground-truth disparity information of $\mathcal{S}$ is provided by the dataset, i. e., $d_{min} = -2.2$, $d_{max} = 1.4$ and $d_{range} = 3.6$ pixels. The first sparsely-sampled EPI of $\mathcal{S}$, represented by ε, is illustrated in Figure 4.6 (a). It can be seen that the shape of ε is 512×9 pixels.

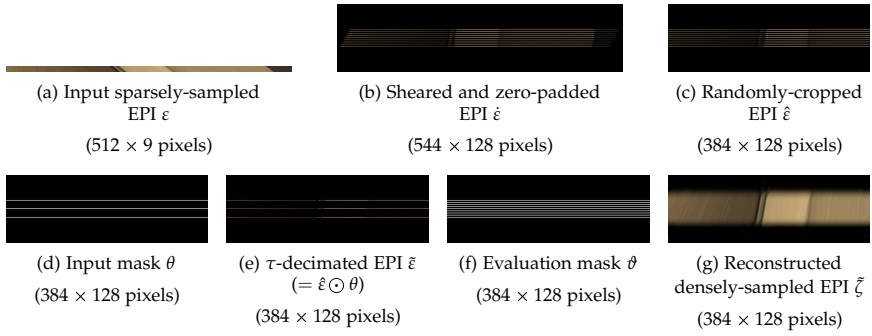


Figure 4.6. Training data preparation and result demonstration. A sparsely-sampled EPI ε from a training 3D SSLF is illustrated in (a). The sheared and zero-padded EPI $\dot{\varepsilon}$ in (b) is the result of performing the pre-shearing and zero-padding step on ε. A random cropping operation is then performed on $\dot{\varepsilon}$ to produce a $3:1$ randomly-cropped EPI $\hat{\varepsilon}$ presented in (c). For $\hat{\varepsilon}$, an input mask θ is designed as shown in (d) and utilized to generate the τ-decimated EPI $\tilde{\varepsilon}$ in (e), which is the input data for the learning-based sparse regularization step of DRST. The evaluation mask ϑ in (f) is employed to calculate the loss function (4.4.12) of the learning-based sparse regularization step. Finally, (g) illustrates the output of the learning-based sparse regularization of DRST, i. e., a reconstructed densely-sampled EPI $\tilde{\zeta}$.

The values of d_{min} and d_{range} are utilized to shear and pad ε as shown in Figure 4.6 (b). Specifically, the sheared and zero-padded EPI $\dot{\varepsilon}$ has nine separated non-black lines from ε. The horizontal and vertical displacements between neighboring non-black lines are φ and $\frac{\tau}{4}$ $(= 4)$, respectively. Here, $\left(d_{min} - \left(\frac{\tau}{4} - d_{range}\right)\right) \leqslant \varphi \leqslant d_{min}$ and $d_{range} \leqslant \frac{\tau}{4}$, such that performing image inpainting on $\dot{\varepsilon}$ can produce a densely-sampled EPI. Moreover, the size of $\dot{\varepsilon}$ is 544×128 pixels. In order to augment training data, an EPI of size of 384×128 pixels, $\hat{\varepsilon}$, is randomly cropped from $\dot{\varepsilon}$ for each training iteration as shown in Figure 4.6 (c). Note that $\dot{\varepsilon}$ and $\hat{\varepsilon}$ have the same height, implying that the random cropping operation here is essentially to randomly slide a $3:1$ window inside $\dot{\varepsilon}$ along the horizontal axis to produce one crop, i. e., $\hat{\varepsilon}$. To ensure the self-supervised learning can be performed in the following learning-based sparse regularization

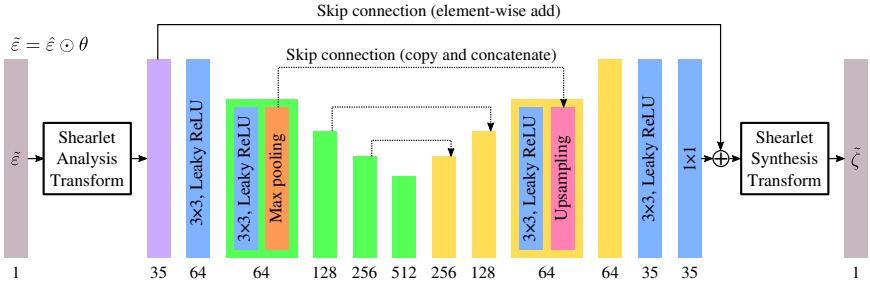


Figure 4.7. Network architecture of the learning-based sparse regularization of DRST.

step, for each randomly-cropped EPI $\hat{\varepsilon}$, we prepare two masks, i. e., input mask θ and evaluation mask ϑ. In particular, the input mask θ has three non-zero lines, corresponding to the first, middle and last non-zero lines of $\hat{\varepsilon}$. The evaluation mask ϑ has nine non-zero lines, corresponding to all the nine non-zero lines of $\hat{\varepsilon}$. The input mask θ and evaluation mask ϑ are illustrated in Figure 4.6 (d) and (f), respectively. In addition, the input mask θ is utilized to generate a sparsely-sampled EPI $\tilde{\varepsilon}$ from the randomly-cropped EPI $\hat{\varepsilon}$ as the input data for the next learning-based sparse regularization step, i. e., $\tilde{\varepsilon} = \hat{\varepsilon} \odot \theta$. The vertical displacement between any two adjacent non-zero lines of the input mask θ or sparsely-sampled $\tilde{\varepsilon}$ is equal to the sampling interval τ ($= 16$). As a result, $\tilde{\varepsilon}$ is also referred to as τ-decimated EPI, which is illustrated in Figure 4.6 (e).

(iii) **Learning-based sparse regularization**. The learning-based sparse regularization is also targeted at solving the optimization problem in Equation 4.4.5. We try to achieve this goal by employing a CNN consisting of an encoder-decoder network and a residual learning strategy, which are originally proposed in U-Net [RFB15] and ResNet [HZR+16], respectively. The network architecture of this CNN is displayed in Figure 4.7, where the design of the encoder-decoder network is inspired by two of the most successful applications of U-Net in computer vision, i. e., Pix2Pix [IZZ+17] for image-to-image translation and Noise2Noise [LMH+18] for photographic noise removal. It

can also be seen from the figure that the input data is the τ-decimated EPI $\tilde{\varepsilon}$ and the output data is the reconstructed densely-sampled EPI $\tilde{\zeta}$, which is also illustrated in Figure 4.6 (g). The shearlet analysis transform $\mathcal{SH}(\cdot)$ converts the one-channel $\tilde{\varepsilon}$ into 35 - channels shearlet coefficients in shearlet domain. These coefficients are then fed to the encoder-decoder network to predict shearlet coefficient residuals. Specifically, the encoder-decoder network is an adapted U-Net with an encoder and a decoder, of which each has χ hierarchies. Each hierarchy in the encoder part is composed of three layers, i. e., a 2D convolution layer, a Leaky ReLU layer ($\alpha = 0.3$) and a max pooling layer for decreasing the spatial resolution by two. Each hierarchy in the decoder part also consists of three layers, i. e., a 2D convolution layer, a Leaky ReLU layer ($\alpha = 0.3$) and an upsampling layer using nearest interpolation for increasing the spatial resolution by two. It is worth mentioning that, since the shearlet analysis and synthesis transforms involve 2D Discrete Fourier Transforms (DFTs) and inverse DFTs that require a significant amount of GPU memory, χ is set to four here to limit the number of trainable parameters. The encoder and decoder in the U-Net are connected by three skip connections (copy and concatenate) at the same spatial resolution for the first three hierarchies. The convolution kernel size is set to 3×3 for all the 2D convolution layers except for the last one, where the convolution kernel size is set to 1×1. In addition, no Leaky ReLU layer is placed after the last 2D convolution layer. The residual learning strategy mitigates the problem of vanishing/exploding gradients [HZR+16] during training, which has been demonstrated effective in recovering coefficients in the contourlet transform domain [DZD06] for limited-angle Computed Tomography (CT) reconstruction [GY17]. Moreover, in order to reconstruct the shearlet coefficients of a target densely-sampled EPI, one requires to amplify the intensities of the shearlet coefficients of the input sparsely-sampled EPI with performing aliasing removal at the same time [VBG18; CTC+00]. A deep learning-based algorithm using the residual learning strategy only predicts the differences between shearlet coefficients of the input and target EPIs, which, in general, enables a better densely-sampled EPI reconstruction performance compared to the manner of utilizing the

same algorithm to predict the complete shearlet coefficients of the target EPIs. Therefore, we adopt the residual learning strategy and add the predicted shearlet coefficient residuals back to the source shealet coefficients by means of the other type of skip connection, i. e., an element-wise add operation. Finally, these processed shearlet coefficients are transformed back to image domain to generate $\tilde{\zeta}$ via the shearlet synthesis transform $\mathcal{SH}^*(\cdot)$. Mathematically, the learning-based sparse regularization can be written as

$$\tilde{\zeta} = \mathcal{SH}^* \Big(\mathcal{SH}(\tilde{\varepsilon}) + \mathcal{R}\big(\mathcal{SH}(\tilde{\varepsilon})\big) \Big), \tag{4.4.10}$$

where $\mathcal{R}(\cdot)$ denotes the encoder-decoder network. For the training case in Figure 4.6, $\mathcal{R} : \mathbb{R}^{384\times128\times35} \rightarrow \mathbb{R}^{384\times128\times35}$. The trainable parameters in $\mathcal{R}$ can be learned by solving the below optimization problem:

$$\min_{\mathcal{R}} \left\| \tilde{\zeta} - \tilde{\zeta}^{\mathrm{GT}} \right\|_1 . \tag{4.4.11}$$

However, the ground-truth densely-sampled EPI $\tilde{\zeta}^{\mathrm{GT}}$ corresponding to the reconstructed densely-sampled EPI $\tilde{\zeta}$ is unknown, since the training synthetic SSLF dataset does not offer the corresponding ground-truth DSLF data. Besides, rendering a high-quality and high-resolution synthetic DSLF dataset is prohibitively expensive compared to the rendering of a synthetic SSLF dataset. Therefore, a new loss function without relying on the DSLF data is proposed by using self-supervised learning. Specifically, the loss function for the training of the encoder-decoder network in the learning-based sparse regularization takes account of minimizing the reconstruction error between the ground-truth sparsely-sampled EPI $\hat{\varepsilon}$ and the reconstructed densely-sampled EPI $\tilde{\zeta}$ using the evaluation mask ϑ via ℓ_1 norm, i. e.,

$$\mathcal{L} = \left\| \hat{\varepsilon} - \tilde{\zeta} \odot \vartheta \right\|_1 . \tag{4.4.12}$$

Although the ground-truth sparsely-sampled EPI $\hat{\varepsilon}$ is not densely-sampled, it contains 6 non-zero lines that the input τ-decimated EPI $\tilde{\varepsilon}$ does not have, thereby guiding the optimization process for the training of the network of the learning-based sparse regularization.

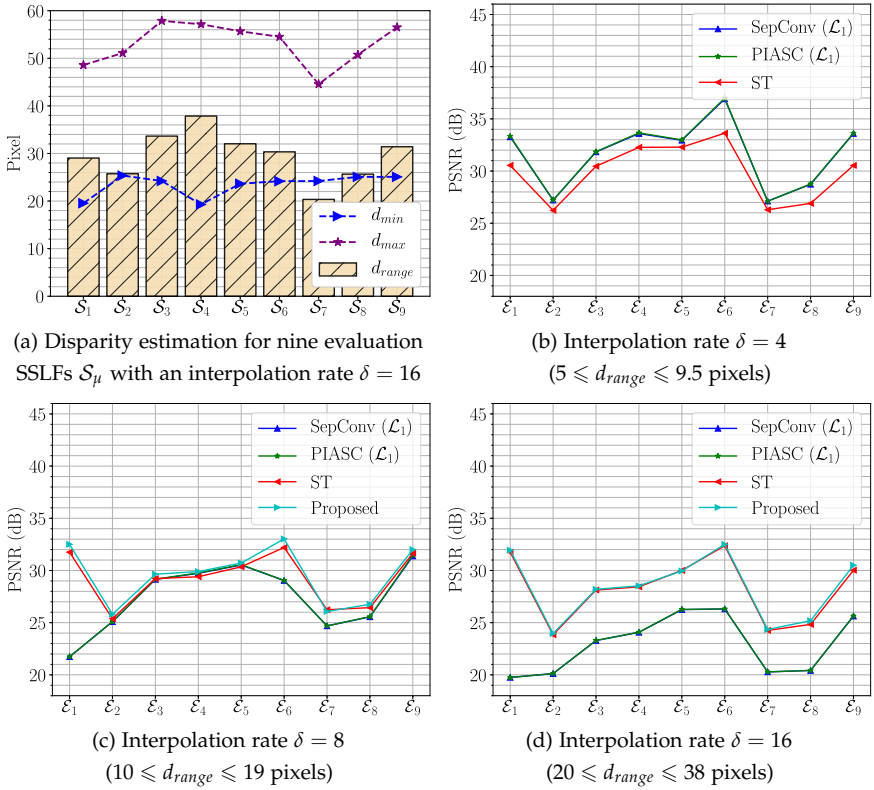
(iv) **Post-shearing**. The post-shearing step of DRST is the same as that of ST as described in Section 4.4.1 (iv). In addition, it is worth mentioning that the post-shearing operation is only performed in the prediction (or evaluation) phase of DRST. The EPI $\tilde{\zeta}$ outputted by the above learning-based sparse regularization step is sheared via the post-shearing operation to produce the final reconstructed densely-sampled EPI ζ.

4.5 Fusion of novel view synthesis and EPI inpainting

The approaches introduced in Section 4.3 and Section 4.4 solve the 3D DSLF reconstruction problem based on the two different representations of the input 3D SSLF $\mathcal{S}$. Specifically, the novel view synthesis-based methods consider $\mathcal{S}$ as a sequence of parallax images, i. e., $\mathcal{S} = \{\mathcal{I}_j | 1 \leqslant j \leqslant n\}$; however, the EPI inpainting-based algorithms treat $\mathcal{S}$ as a set of sparsely-sampled EPIs, i. e., $\mathcal{S} = \{\varepsilon_i | 1 \leqslant i \leqslant l\}$. Depending on the varying disparity conditions of the input 3D SSLFs, the 3D DSLF reconstruction performance of these two categories of methods is different. Therefore, this section focuses on investigating how to effectively and efficiently combine these two kinds of 3D DSLF reconstruction approaches to achieve a better 3D DSLF reconstruction performance than using either of them alone.

4.5.1 Interpolation-Enhanced Shearlet Transform (IEST)

Although ST is a universal solution to the light field reconstruction problem on 3D SSLFs with varying disparities, it is not as effective as one of the state-of-the-art video frame interpolation methods, i. e., SepConv, for light field reconstruction from SSLFs with small disparity ranges. An example for this phenomenon is given in Figure 4.8 (b), where the interpolation rate $\delta = 4$ equals to $5 \leqslant d_{range} \leqslant 9.5$ pixels, derived from $20 \leqslant d_{range} \leqslant 38$ pixels in the case of $\delta = 16$ in Figure 4.8 (a), for all the evaluation 3D SSLFs $\mathcal{S}_\mu$ ($1 \leqslant \mu \leqslant 9$) in [GKB+19b]. Similar to the sampling interval τ in Section 4.2, the interpolation rate δ also represents the number of views to be interpolated between neighboring views in $\mathcal{S}_\mu$. However, the difference



(a) Disparity estimation for nine evaluation SSLFs $\mathcal{S}_\mu$ with an interpolation rate $\delta = 16$

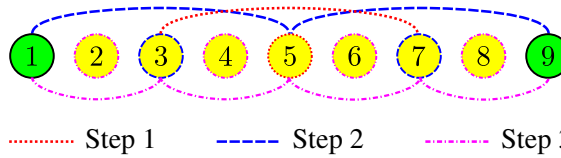
(b) Interpolation rate $\delta = 4$ ($5 \leqslant d_{range} \leqslant 9.5$ pixels)

(c) Interpolation rate $\delta = 8$ ($10 \leqslant d_{range} \leqslant 19$ pixels)

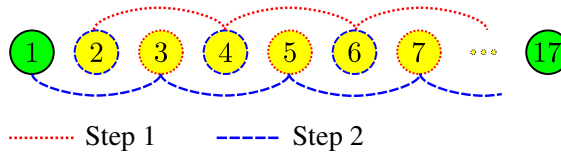
(d) Interpolation rate $\delta = 16$ ($20 \leqslant d_{range} \leqslant 38$ pixels)

Figure 4.8. Minimum per-view PSNR results (in dB, explained in Section 4.5.1) of different light field reconstruction methods on nine evaluation datasets with different interpolation rates $\delta \in \{4, 8, 16\}$. (Source: [GKB+19b])

between them is that $\delta \leqslant \tau$ and $\tau \% \delta = 0$; in other words, the number of the parallax views of the reconstructed desired 3D light field $\mathcal{E}_\mu$, i. e., $\big((n-1)\delta + 1\big)$, may be less than that of $\mathcal{D}_\mu$, i. e., $\big((n-1)\tau + 1\big)$. It can also be found that $\mathcal{E}_\mu$ will not be densely-sampled if $\delta < d_{range}$. Besides, the minimum per-view PSNR of the reconstructed desired 3D light field



(a) Interpolation rate $\delta = 8$ (for moderate disparity range, i. e., 8 - 16 pixels)



(b) Interpolation rate $\delta = 16$ (for large disparity range, i. e., > 16 pixels)

Figure 4.9. Flowcharts of IEST for light field reconstruction from SSLFs at different interpolation rates, i. e., $\delta \in \{8, 16\}$. (Source: [GKB+19b])

$\mathcal{E}_\mu$ is used as the evaluation criterion[3] in Figure 4.8 (b), where the novel view synthesis-based methods, i. e., SepConv and PIASC, achieve better performance than the EPI inpainting-based method, i. e., ST. However, for input SSLFs with larger disparity ranges, ST tends to be more effective than SepConv as shown in Figure 4.8 (c) and (d). Intuitively, taking advantage of SepConv to refine the parallax views of light fields that are reconstructed by ST from SSLFs with moderate and large disparity ranges may improve the final light field reconstruction performance. Therefore, a novel light field reconstruction method, i. e., IEST [GKB+19b], is proposed. The IEST method is specially designed for light field reconstruction on 3D SSLFs with moderate and large disparity ranges with a consideration that the reconstructed parallax views of ST involving small disparity ranges can be refined by SepConv. Depending on different interpolation rates, two parallax view refinement strategies of IEST are presented in Figure 4.9. As shown in (a), the first strategy is designed for the case of interpolation rate $\delta = 8$. Here, green circles stand for the ground-truth parallax views from an input 3D SSLF $\mathcal{S}$ (also see Figure 4.2 (a)) and yellow circles denote the

[3]Here, the ground-truth light field $\mathcal{E}_\mu^{\mathrm{GT}}$ corresponding to the reconstructed target 3D light field $\mathcal{E}_\mu$ is known. The per-view PSNR calculation is performed for images in $\mathcal{E}_\mu^{\mathrm{GT}} \backslash \mathcal{S}_\mu$.

parallax views reconstructed by ST. These reconstructed parallax views are then refined by SepConv recursively, which is depicted by using three types of dash lines that represent Steps 1, 2 and 3. To be precise, each step leverages two parallax views having a small disparity range to synthesize the intermediate view between them. For the interpolation rate $\delta = 16$, the second strategy of IEST refines the parallax views reconstructed by ST by means of only two steps as shown in (b). It is worth mentioning that the strategy of IEST designed for $\delta = 8$ is especially effective for the light field reconstruction on SSLFs with moderate disparity ranges (8 - 16 pixels), while the second IEST strategy designed for $\delta = 16$ is more effective for the light field reconstruction on SSLFs with large disparity ranges (> 16 pixels).

4.5.2 Flow-Assisted Shearlet Transform (FAST)

The previous subsection about IEST describes how to reconstruct a target 3D light field $\mathcal{E}$ from an input 3D SSLF $\mathcal{S}$ in a coarse-to-fine manner using both ST and SepConv. The main drawback of IEST is that the interpolation rate δ of $\mathcal{E}$ should be a power of two, which is caused by SepConv that can hardly handle an arbitrary sampling interval τ (More details are explained in Section 4.3.3); in other words, for the light field reconstruction tasks that require an arbitrary δ, IEST tends to fail. To overcome this limitation, this subsection will introduce a universal solution to the 3D DSLF reconstruction problem using optical flow techniques. Specifically, inspired by the success of Super-SloMo [JSJ+18] in video frame interpolation, a novel learning-based method, referred to as FAST [GKB+19a], is proposed to reconstruct 3D DSLFs from 3D SSLFs. The FAST method adopts one of the state-of-the-art optical flow approaches, i. e., PWC-Net [SYL+18], to estimate the bidirectional optical flow between neighboring views in an input SSLF. Besides, FAST also leverages one of the state-of-the-art DSLF reconstruction methods, i. e., ST, to guide novel view synthesis. The architecture of FAST is illustrated in Figure 4.10. It can be seen that FAST is composed of two CNNs based on the U-Net architecture, i. e., DRN and VSN. Regarding the architecture of DRN, it has six hierarchies in the encoder part and five hierarchies in the decoder part with the same architecture as the flow interpolation CNN in Super-SloMo. Since the horizontal-parallax

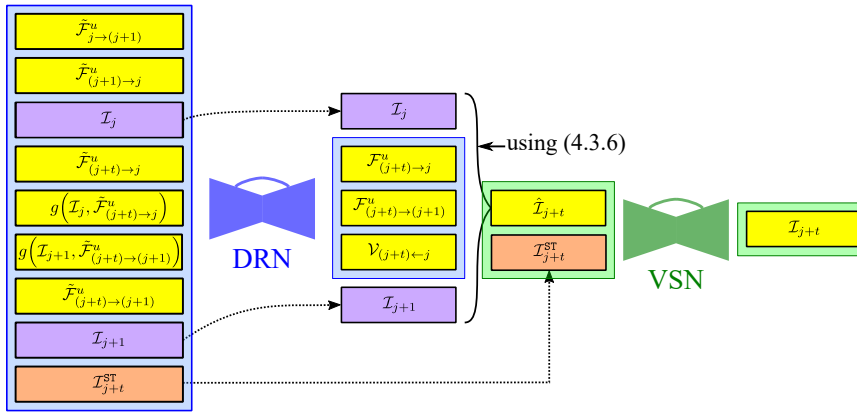


Figure 4.10. Network architecture of FAST. $\mathcal{I}^{\mathrm{ST}}_{j+t}$ is the view reconstructed by ST, where $t \in \left\{\frac{1}{\delta}, \frac{2}{\delta}, ..., \frac{\delta-1}{\delta}\right\}$ and δ is the interpolation rate. (Source: [GKB+19a])

SSLF shown in Figure 4.2 (a) does not involve any object motion along the vertical axis, only the horizontal components of the bidirectional optical flow estimated by PWC-Net contain useful information, which are represented by bidirectional disparity maps $\tilde{\mathcal{F}}^u_{j\to(j+1)}$ and $\tilde{\mathcal{F}}^u_{(j+1)\to j}$. The inverse disparity maps, i. e., $\tilde{\mathcal{F}}^u_{(j+t)\to j}$ and $\tilde{\mathcal{F}}^u_{(j+t)\to(j+1)}$, can then be estimated by using Equation 4.3.8. The DRN of FAST takes $\tilde{\mathcal{F}}^u_{j\to(j+1)}$, $\tilde{\mathcal{F}}^u_{(j+1)\to j}$, $\tilde{\mathcal{F}}^u_{(j+t)\to j}$, $\tilde{\mathcal{F}}^u_{(j+t)\to(j+1)}$, $g\left(\mathcal{I}_j, \tilde{\mathcal{F}}^u_{(j+t)\to j}\right)$, $g\left(\mathcal{I}_{j+1}, \tilde{\mathcal{F}}^u_{(j+t)\to(j+1)}\right)$, $\mathcal{I}_j$, $\mathcal{I}_{j+1}$ and $\mathcal{I}^{\mathrm{ST}}_{j+t}$ as the input (19 channels in total) and outputs $\mathcal{F}^u_{(j+t)\to j}$, $\mathcal{F}^u_{(j+t)\to(j+1)}$ and $\mathcal{V}_{(j+t)\leftarrow j}$, which are used to interpolate an intermediate view $\hat{\mathcal{I}}_{j+t}$ via Equation 4.3.6. With regard to the architecture of VSN, it is a "shallow" version of DRN with four hierarchies in the encoder part and three hierarchies in the decoder part. The interpolated novel view $\hat{\mathcal{I}}_{j+t}$ and the corresponding view reconstructed by ST, i. e., $\mathcal{I}^{\mathrm{ST}}_{j+t}$, are fed to the VSN of FAST to generate the final target view $\mathcal{I}_{j+t}$. Note that $t \in \left\{\frac{1}{\delta}, \frac{2}{\delta}, ..., \frac{\delta-1}{\delta}\right\}$ and $\mathcal{I}_{j+t}$ corresponds to the image $\tilde{\mathcal{I}}_{(j-1+t)\tau+1}$ of the target 3D DSLF $\mathcal{D}$ (cf. Section 4.3.4). The loss function of FAST is composed of VSN reconstruction loss, DRN reconstruction loss and warping loss, of which all are

based on ℓ_1 norm:

$$\mathcal{L}^{\mathsf{FAST}} = \omega_1 \mathcal{L}^{\mathsf{VSN}} + \omega_2 \mathcal{L}^{\mathsf{DRN}} + \omega_3 \mathcal{L}^{\mathsf{W}}, \tag{4.5.1}$$

where

$$\begin{aligned} \mathcal{L}^{\mathsf{VSN}} = \left\| \mathcal{I}_{j+t} - \mathcal{I}^{\mathsf{GT}}_{j+t} \right\|_1, \mathcal{L}^{\mathsf{DRN}} = \left\| \hat{\mathcal{I}}_{j+t} - \mathcal{I}^{\mathsf{GT}}_{j+t} \right\|_1, \\ \mathcal{L}^{\mathsf{W}} = \left\| g\left(\mathcal{I}_j, \mathcal{F}^u_{(j+t)\to j}\right) - \mathcal{I}^{\mathsf{GT}}_{j+t} \right\|_{j+1} + \left\| g\left(\mathcal{I}_{j+1}, \mathcal{F}^u_{(j+t)\to(j+1)}\right) - \mathcal{I}^{\mathsf{GT}}_{j+t} \right\|_1, \end{aligned} \tag{4.5.2}$$

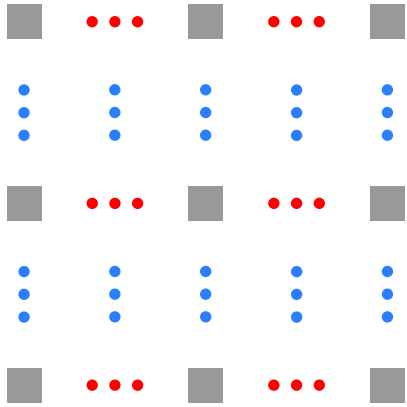
$\omega_1 = 9$, $\omega_2 = 2$ and $\omega_3 = 1$. Note that these weights are set empirically with a consideration that the VSN reconstruction loss is more important than the other two losses.

4.6 4D DSLF reconstruction

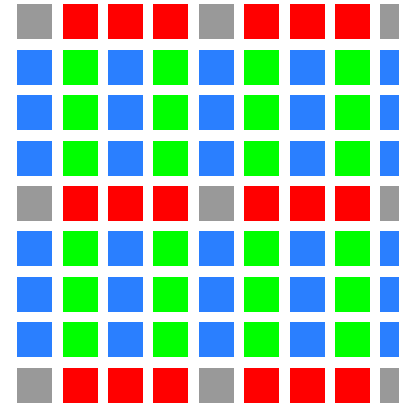
After introducing the two categories of solutions to 3D DSLF reconstruction, i. e., novel view synthesis and EPI inpainting in Section 4.3 and Section 4.4, respectively, and the fusion of them in Section 4.5, in this section we focus on studying how to take full advantage of all these 3D DSLF reconstruction solutions to solve the 4D DSLF reconstruction problem. Depending on the desired sampling interval τ, two strategies of using 3D DSLF reconstruction algorithms to solve the 4D DSLF reconstruction problem are illustrated in Figure 4.11 (a) and (b), respectively. In both of them, the gray blocks stand for the full-parallax views from an input 4D SSLF. More details about these two strategies (one for the general case and the other for the special case) are discussed in the next two subsections.

4.6.1 Strategy for the general case

The 4D DSLF reconstruction problem is similar to the 3D DSLF problem discussed in Section 4.2. The only difference between them is that, in a 3D SSLF, the parallax views are evenly distributed along the axis 's' or 't' (see Figure 4.1), while in a 4D SSLF, the parallax images are uniformly sampled along both axes. Given an input 4D SSLF $\mathcal{S}^{\mathsf{4D}}$ with $n \times n$ full-parallax views, the goal of 4D DSLF reconstruction is to generate a desired target 4D DSLF $\mathcal{D}^{\mathsf{4D}}$ with $\dot{n} \times \dot{n}$ full-parallax views. The relationship between n



(a) Strategy one designed for the general case (τ is arbitrary)



(b) Strategy two designed for the special case (τ is a power of two)

Figure 4.11. Two strategies of 4D DSLF reconstruction using 3D DSLF reconstruction methods. For an arbitrary sampling interval τ in (a), a 4D DSLF is reconstructed from a 4D SSLF in two steps, of which the results are represented by red and blue dots, respectively. For a special sampling interval $\tau = \{4, 8, 16, 32, \ldots\}$ in (b), the 4D DSLF reconstruction is performed via three steps, of which the results are represented by red, blue and green blocks, respectively.

and $\dot{n}$ is determined by the sampling interval τ, i. e., $\dot{n} = \big((n-1)\tau + 1\big)$, as explained in Section 4.2. In terms of the general case that a 4D DSLF reconstruction task requires an arbitrary sampling interval τ, several 3D DSLF reconstruction methods can be adopted to solve this task. Specifically, the ST, MAST and FAST approaches can reconstruct the desired target 4D DSLF $\mathcal{D}^{4D}$ from the given input 4D SSLF $\mathcal{S}^{4D}$ in two steps using the strategy illustrated in Figure 4.11 (a).

(i) As can be seen from this figure, the input $\mathcal{S}^{4D}$ is first split into n ($= 3$ in the given example) horizontal-parallax SSLFs. After this, any of the aforementioned three different 3D DSLF reconstruction approaches can be applied to these horizontal-parallax SSLFs to generate the corresponding horizontal-parallax DSLFs, which are represented by red dots and gray blocks.

(ii) These generated 3D DSLFs are then treated as $\dot{n}$ vertical-parallax SSLFs. The same 3D DSLF reconstruction method is utilized to recover the missing $\dot{n} \cdot (\dot{n} - n)$ views, which are denoted by the blue dots in the figure.

4.6.2 Strategy for the special case

In the previous subsection about the 4D DSLF reconstruction strategy for the general case that the sampling interval τ is an arbitrary number, the SepConv, PIASC, DRST and IEST methods can hardly be applied, mainly because these methods require τ to be a power of two (see Section 4.3.3). This subsection will introduce how to leverage all the seven different 3D DSLF reconstruction methods mentioned in this chapter, i. e., SepConv, PIASC, ST, MAST, DRST, IEST and FAST, to reconstruct a desired target 4D DSLF $\mathcal{D}^{4D}$ from an input 4D SSLF $\mathcal{S}^{4D}$ in a special case that the sampling interval τ is a power of two, i. e., $\tau = \{4, 8, 16, 32, \ldots\}$. The strategy of using all the 3D DSLF reconstruction methods to solve the 4D DSLF reconstruction problem for the special case here is displayed in Figure 4.11 (b). As shown in this figure, this strategy is composed of three steps originally proposed in [VBG18].

(i) The input $\mathcal{S}^{4D}$ is first considered as n horizontal-parallax 3D SSLFs. Any of the seven different 3D DSLF reconstruction methods can then be used to reconstruct the $n \cdot (\dot{n} - n)$ missing views in $\mathcal{D}^{4D}$, which are represented by the red blocks in the figure.

(ii) Subsequently, the reconstructed 3D DSLFs are converted into $\dot{n}$ vertical-parallax SSLFs. The vertical-parallax DSLFs in odd columns of $\mathcal{D}^{4D}$, i. e., $s = \{1, 3, 5, \ldots, (\dot{n} - 2), \dot{n}\}$, are then reconstructed from the corresponding vertical-parallax SSLFs. The reconstructed views in these vertical-parallax DSLFs are indicated by blue blocks, of which the number is $\frac{\dot{n}+1}{2}(\dot{n} - n)$.

(iii) Finally, the partially-reconstructed $\mathcal{D}^{4D}$ is split into $\dot{n}$ horizontal-parallax SSLFs, to which the same 3D DSLF reconstruction method is applied. The rest of the unknown views in $\mathcal{D}^{4D}$ is therefore recovered as indicated by the green blocks in Figure 4.11 (b), of which the number is $\frac{\dot{n}-1}{2}(\dot{n} - n)$.

4.6.3 Discussions

The above two subsections introduce how to use two strategies consisting of different 3D DSLF reconstruction methods to perform 4D DSLF reconstruction on 4D SSLF data w.r.t. the general and special cases. It is worth mentioning that, for 4D DSLF reconstruction in the special case, the ST-based 3D DSLF reconstruction approaches, i. e., ST, MAST, DRST, IEST and FAST, perform more efficiently using the strategy two than using the strategy one. In particular, in the two steps of the first strategy designed for the general case, only one shearlet system in ST is constructed, since the disparity ranges of the horizontal-parallax SSLFs in step (i) are the same as those of the vertical-parallax SSLFs in step (ii). However, in the second strategy designed for the special case, the shearlet system constructed in steps (i) and (ii) is different from the shearlet system constructed in step (iii). Specifically, the disparity ranges of the horizontal-parallax SSLFs in step (iii) are half of those of horizontal-parallax SSLFs in step (i) and vertical-parallax SSLFs in step (ii). Therefore, the shearlet system in step (iii) has one scale less than the shearlet system used in steps (i) and (ii). It can be seen from Section 4.4.1 (ii) and (iii) that the less scales in the constructed shearlet system, the less time will be required to perform ST on 3D SSLF data. It is therefore suggested that, for 4D DSLF reconstruction in the special case that τ is a power of two, the strategy two is preferable than the strategy one.

Chapter 5

Conclusions

5.1 Summary

In this thesis, we present a novel movable 1D large-scale light field acquisition system for capturing either horizontal-parallax or full-parallax SSLFs for real-world scenes. This system contains 24 RGB cameras with large baselines (≈ 11 cm) and two Kinect V2 sensors with a large displacement (≈ 2.4 m). In order to produce high-quality and high-fidelity DSLF data for VR, 3DTV and holographic devices using our large-scale light field acquisition system, we mainly focus on studying how to overcome the following three challenges:

(i) The estimation of the rigid transformation from the coordinates of one Kinect V2 to the coordinates of each RGB camera;

(ii) The estimation of the rigid transformation from the coordinates of one Kinect V2 to the coordinates of the other;

(iii) The effective and efficient DSLF reconstruction on the SSLF data with moderate disparity ranges (8 - 16 pixels) or large disparity ranges ($>$ 16 pixels).

To solve the challenging problem (i), we propose a novel RGB-Kinect calibration method based on the coarse-to-fine framework, composed of the coarse estimation and estimation refinement parts that exploit the geometric constraints in the scene and cameras, but no calibration objects, e. g., checkerboards. In the coarse estimation part, a camera orientation approximation method is leveraged to estimate the rigid transformations from the coordinates of a Kinect V2 camera to the coordinates of its 12 nearest RGB cameras. The coarsely-estimated rigid transformations are

refined in the estimation refinement stage using a state-of-the-art BA method.

To address the challenging problem (ii), a novel Kinect registration approach in a coarse-to-fine fashion is proposed by leveraging the local color and geometry information, but no calibration objects. In the coarse estimation stage, the off-the-shelf feature detector SURF and the CNN-based 3D descriptor 3DMatch, are employed to describe the local color and geometry information, respectively. Both color and geometry descriptors are exploited to estimate an initial rough rigid transformation between two Kinect V2 cameras, which can be refined by an ICP-based algorithm in the estimation refinement stage.

To resolve the challenging problem (iii), we propose three groups of 3D DSLF reconstruction methods, i. e., novel view synthesis, EPI inpainting and the fusion of them. In terms of the novel view synthesis group, a novel parallax view generation algorithm, PIASC, based on the state-of-the-art video frame interpolation method, SepConv, is proposed to reconstruct a DSLF from an input SSLF in a recursive manner. Regarding the EPI inpainting group, two methods, MAST and DRST, based on the state-of-the-art DSLF reconstruction method, ST, are proposed to reconstruct a densely-sampled EPI for an input sparsely-sampled EPI. In addition, two approaches, IEST and FAST, belonging to the group of the fusion of novel view synthesis and EPI inpainting, are proposed to improve the DSLF reconstruction performance of ST. Finally, we present two strategies for solving the 4D DSLF reconstruction problem using the 3D DSLF reconstruction methods.

5.2 Future work

The research work in this thesis can be extended in the following directions:

(i) The DSLF reconstruction method ST for EPI inpainting relies on a specifically-tailored 2D shearlet system. It would be interesting to construct a specifically-tailored 3D shearlet system for ST algorithm to perform light field angular resolution enhancement directly on the input 3D SSLF volume as shown in Figure 4.2 (b).

(ii) The ST-based DSLF reconstruction methods in this thesis are successful in light field reconstruction for Lambertian scenes and non-Lambertian scenes consisting of semi-transparent objects. However, ST tends to fail in light field reconstruction for non-Lambertian scenes containing highly complex specular geometry. It would be interesting to combine ST with the current most popular scene representation MPI under a learning-based framework for DSLF reconstruction and novel view synthesis;

(iii) The DRST approach is initially designed for DSLF reconstruction on SSLFs with moderate disparity ranges (8 - 16 pixels). It would be interesting to change the trunk network of DRST from U-Net to convolutional Long Short-Term Memory (LSTM) [SCW+15] to handle the case of large disparity ranges (> 16 pixels);

(iv) The ST-based methods rely on the handcrafted shearlet filters. It would be interesting to automatically learn or refine these filters from the training data in a data-driven and deep learning-based manner;

(v) For light field angular super-resolution, the ST-based DSLF reconstruction is one of the current state-of-the-arts. It would be interesting to combine ST with the state-of-the-art learning-based image super-resolution methods to handle light field angular and spatial super-resolution at the same time.

Chapter 6

Publications

6.1 Publication 1

A Linear Method for Recovering the Depth of Ultra HD Cameras Using a Kinect V2 Sensor

Yuan Gao, Matthias Ziegler, Frederik Zilly, Sandro Esquivel and Reinhard Koch

Published in

2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), Nagoya, Japan, 8-12 May 2017, pages 494-497.

DOI: 10.23919/MVA.2017.7986908

A Linear Method for Recovering the Depth of Ultra HD Cameras Using a Kinect V2 Sensor

Yuan Gao*† Matthias Ziegler† Frederik Zilly† Sandro Esquivel* Reinhard Koch*
*Institute of Computer Science, Christian-Albrechts-University of Kiel, 24118 Kiel, Germany
{yga, sae, rk}@informatik.uni-kiel.de
†Fraunhofer Institute for Integrated Circuits IIS, 91058 Erlangen, Germany
{matthias.ziegler, frederik.zilly}@iis.fraunhofer.de

Abstract

Depth-Image-Based Rendering (DIBR) is a mature and important method for making free-viewpoint videos. As for the study of the DIBR approach, on the one hand, most of current research focuses on how to use it in systems with low resolution cameras, while a lot of Ultra HD rendering devices have been launched into markets. On the other hand, the quality and accuracy of the depth image directly affects the final rendering result. Therefore, in this paper we try to make some improvements on solving the problem of recovering the depth information for Ultra HD cameras with the help of a Kinect V2 sensor. To this end, a linear least squares method is proposed, which recovers the rigid transformation between a Kinect V2 and an Ultra HD camera, using the depth information from the Kinect V2 sensor. In addition, a non-linear coarse-to-fine method, which is based on Sparse Bundle Adjustment (SBA), is compared with this linear method. Experiments show that our proposed method performs better than the non-linear method for the Ultra HD depth image recovery both in computing time and precision.

1 Introduction

With so many Ultra HD resolution 3D TVs and high resolution Virtual Reality (VR) headsets having been launched into the market, the creation of the high-quality and high-resolution contents for these devices is becoming a research hotspot. Depth-Image-Based Rendering (DIBR) [1] is such a method which can be used for free-viewpoint video creation. The depth information in DIBR is very important because the accuracy of it is the key influence factor for the quality of free-viewpoint rendering. Therefore, in this paper, we focus on the depth information recovery for Ultra HD RGB cameras. The depth recovery for multiple RGB cameras has been well researched and can be classified into two categories. One is the light field-to-depth approach [2, 3], the other is stereo matching [4, 5].

Since most of the color-image-based methods above may not address the problem of recovering the depth information of regions without textures, it is also a good choice to solve this problem using the depth information from depth sensors. To achieve this, the calibration between the depth sensor and color cameras is a crucial step. Zhang *et al.* propose a maximum likelihood solution for this calibration problem using a Kinect V1 [6]. However, the distortion of the depth values is not addressed by their method. Herrera *et al.* propose a calibration algorithm for a Kinect V1 depth sensor and a color camera pair with distortion correction [7]. Hansard *et al.* find a 3D projective transformation for the ToF-stereo calibration of a time-of-flight (ToF) sensor and two RGB cameras [8]. Jung *et al.* design a special 2.5D pattern board for the calibration of a low resolution ToF sensor and a high resolution RGB camera [9].

The second version of the Microsoft Kinect (Kinect V2) is also based on the ToF technology and is one of the most high-speed and low-cost ToF sensors in the market. Besides, the difference between Kinect V2 and Kinect V1 is well studied in [10, 11], where it is stated that the Kinect V2 has higher accuracy than Kinect V1.

In this paper, we try to make full use of the ToF sensor in a Kinect V2 camera to map the depth information to an Ultra HD resolution camera. To this end, a linear least squares method is proposed. Specifically, a regular 2D checkerboard is employed to find corresponding points between the Kinect V2 sensor and the Ultra HD camera. Then, the rigid transformation between these two cameras is solved by the least squares method. Furthermore, a non-linear coarse-to-fine solution is also explored and compared with the linear one. The difference between the non-linear approach and the metric calibration method in [12] is that the corner points in the 3D space are recovered through the Kinect V2 sensor. Experiments are conducted on a camera rig with one Kinect V2 and one Sony DSLR camera. Experimental results show the superiority of the proposed linear method both in precision and computing time.

2 Methodology

In this section, the calibration process of a Kinect V2 camera and an Ultra HD camera is introduced in detail.

2.1 Preliminary

The internal parameters of pinhole cameras are important properties for camera calibration. To approximate these factors, substantial methods have been proposed [13]. For the Ultra HD camera in our system, the traditional checkerboard-based method is adopted [14]. For the Kinect V2 camera, the ToF sensor in it can also be modeled as a pinhole camera [15]. Its intrinsic parameters can be accessed through the Kinect for Windows SDK or computed in the same way as for a color camera. These intrinsic parameters are then used to compensate lens distortions of both cameras.

2.2 Linear Method

Suppose a pair of corresponding points in the Kinect V2 and camera image planes is measured by the checkerboard corner-based method. The point in the

3D coordinate system of the Kinect V2 is given as $\boldsymbol{x}_i = [x_i \quad y_i \quad z_i]^{\mathrm{T}}$.The corresponding point in the Ultra HD camera image plane is denoted as $\boldsymbol{u}_i = [u_i \quad v_i \quad 1]^{\mathrm{T}}$ in the homogeneous coordinates. The 3D point $\boldsymbol{x}_i$ is first transferred to the camera coordinate system of the Ultra HD camera using the rigid transformation defined by a rotation $\mathbf{R}$ and a translation $\boldsymbol{t}$, then projected to the image coordinate system of this Ultra HD camera. Therefore, the transformation and projection process can be described as:

$$\mathbf{K}(\mathbf{R}\boldsymbol{x}_i + \boldsymbol{t}) = \lambda \boldsymbol{u}_i \tag{1}$$

Here, $\mathbf{K}$ is the camera matrix of the Ultra HD camera and λ is a scaling factor. To simplify equation (1), we use

$$\boldsymbol{p}_i = \mathbf{K}^{-1}\boldsymbol{u}_i \tag{2}$$

on the right side, where $\boldsymbol{p}_i$ is a calibrated image point and $\boldsymbol{p}_i = [p_i \quad q_i \quad 1]^{\mathrm{T}}$. Therefore, equation (1) becomes:

$$\mathbf{R}\boldsymbol{x}_i + \boldsymbol{t} = \lambda \boldsymbol{p}_i \tag{3}$$

The scaling factor λ is calculated from equation (3) as:

$$\lambda = [r_{31} \quad r_{32} \quad r_{33}]\,\boldsymbol{x}_i + t_3 \tag{4}$$

Then, equation (3) can be written as:

$$\begin{bmatrix} \boldsymbol{x}_i^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & -p_i\boldsymbol{x}_i^{\mathrm{T}} & 1 & 0 & -p_i \\ \mathbf{0}^{\mathrm{T}} & \boldsymbol{x}_i^{\mathrm{T}} & -q_i\boldsymbol{x}_i^{\mathrm{T}} & 0 & 1 & -q_i \end{bmatrix} \boldsymbol{h} = \mathbf{0} \tag{5}$$

where

$$\boldsymbol{h} = [r_{11} \quad r_{12} \quad \cdots \quad r_{33} \quad t_1 \quad t_2 \quad t_3]^{\mathrm{T}} \tag{6}$$

Here, $\boldsymbol{h}$ is a vector with twelve variables describing the rigid transformation between the ToF sensor of Kinect V2 and the Ultra HD camera. This problem can be solved by a Linear Least Squares (LLS) method like Singular Value Decomposition (SVD). At least six corresponding point pairs should be utilized to solve it. However, the drawback of this method is that the output $\mathbf{R}$ is not a standard rotation matrix, which is needed for refinement by other parameter estimation methods, *e.g.*, Bundle Adjustment (BA).

2.3 Non-Linear Method

A non-linear method is exploited here in order to make a comparison with the linear one. A coarse-to-fine strategy is adopted to make the method more robust.

2.3.1 Coarse Estimation

Suppose there are a Kinect V2 and an Ultra HD camera. A corresponding point pair in the Kinect V2 3D coordinate and the Utra HD camera image plane are denoted as $\boldsymbol{x}_i$ and $\boldsymbol{u}_i$ as in Section 2.2. The coarse estimation step is designed to give a coarse estimation of the rigid transformation from the Kinect V2 depth sensor coordinate system to the Ultra HD camera coordinate system, which is defined as $(\mathbf{R}, \boldsymbol{t})$. The rigid transformation estimation is obtained by minimizing the following formula:

$$\min_{\mathbf{R},\boldsymbol{t}} \sum_{i=1}^{n} \|\boldsymbol{u}_i - \hat{\boldsymbol{u}}(\boldsymbol{x}_i, \mathbf{K}, \mathbf{R}, \boldsymbol{t})\|^2 \tag{7}$$

Here, n stands for the number of corresponding point pairs. Note that the 3D coordinate system of the ToF sensor in the Kinect V2 camera is used as the reference coordinate system here, which is different from [14], where the 3D coordinates in a model plane are treated as the reference coordinate system. Equation (7) describes a non-linear least squares problem, which can be solved by the Levenberg-Marquardt optimization approach.

2.3.2 Estimation Refinement

After the coarse estimation process, the $\mathbf{R}$ and $\boldsymbol{t}$ for transferring the 3D points from a Kinect V2 ToF sensor to an Ultra HD camera have been computed. To make this problem more general, suppose there is one Kinect V2 camera with multiple Ultra HD cameras, the number of which is denoted as m. The rigid transformation between the Kinect V2 and the Ultra HD camera j is expressed as $\mathbf{R}_j$ and $\boldsymbol{t}_j$. The estimation refinement step can be formulated as:

$$\min_{\mathbf{K}_j,\mathbf{R}_j,\boldsymbol{t}_j} \sum_{j=1}^{m+1} \sum_{i=1}^{n} v_{ij} \|\boldsymbol{u}_{ij} - \hat{\boldsymbol{u}}(\boldsymbol{x}_i, \mathbf{K}_j, \mathbf{R}_j, \boldsymbol{t}_j)\|^2 \tag{8}$$

Here, $\mathbf{K}_j, \mathbf{R}_j, \boldsymbol{t}_j$ with $j = 1, \ldots, m$ relate to the Ultra HD cameras and $\mathbf{K}_{m+1}, \mathbf{R}_{m+1}, \boldsymbol{t}_{m+1}$ to the Kinect V2 sensor. Furthermore, $\boldsymbol{u}_{ij}$ is the ground truth point in the j-th image plane corresponding to a 3D point $\boldsymbol{x}_i$ in the world coordinate system, and $v_{ij} \in \{0, 1\}$ denotes the visibility between these two points. To solve this problem, the 3D coordinate system of the ToF sensor in the Kinect V2 camera is set as the world coordinate system. A generic Sparse Bundle Adjustment (SBA) method is employed to solve this non-linear least squares problem efficiently [16].

(a) Frontal view (b) Vertical view

Figure 1. Cameras in our system.

3 Experiment

3.1 Experimental Settings

System: The system is built on a rig on a tripod, using a Sony α7R II DSLR camera mounted with a Canon lens and a Kinect V2 sensor. The positions and orientations of these two cameras are illustrated in Fig. 1. The displacement between the centers of the two cameras is around 19 centimeters. Note that the original resolution of the Sony camera is $7,952 \times 5,304$ pixels, which is downsampled to the Ultra HD resolution of $3,840 \times 2,561$ pixels for the experimental evaluations described here. The depth sensor in a Kinect V2 has a resolution of 512×424 pixels.

Field of view: The Field of View (FOV) of the depth sensor in the Kinect V2 camera is $\approx$ 70 degrees [17]. Since the focal length of the Sony camera can be adjusted, we set the FOV of the Ultra HD camera to a similar FOV as the Kinect V2 sensor. An example capture of both cameras for the same scene is shown in (a) and (b) in Fig. 2.

Table I. The RMSE and computing time of different methods.

Method	RMSE (pixel)	Time (ms)
Linear Method	0.781	1.2
Non-linear Method (Coarse)	2.045	1.6
Non-linear Method (Refine)	0.993	16.0

Intrinsic parameter: The intrinsic parameters of both the Ultra HD camera and the ToF sensor in the Kinect V2 camera are estimated by a standard checkerboard-based calibration process. The checkerboard used in our experiments has 266 (19×14) corners and the size of each black or white square field is 15×15 mm. The internal parameters are then utilized to undistort all the output views of both cameras.

Corresponding point pair: To evaluate the effects of both methods, corresponding point pairs need to be found in advance. Here, a bigger checkerboard with 54 (9×6) corners is placed in the jointly visible areas of both cameras twice. The size of each square of this checkerboard is 52×52 mm. Therefore, in total 108 corner points are detected automatically in each camera. It should be noted that in the view of the Kinect V2 camera, the infrared view is actually used for detecting the corner points and the depth values of them are estimated by using the same specific filter as described in [18] on the corresponding depth image.

Ultra HD depth recovery: Because there is a significant difference in resolutions of these two cameras, it is prone to get a recovered depth image with most of the information missing by directly performing the rigid transformation from the low-resolution Kinect V2 ToF sensor to the Ultra HD camera image. To solve this problem, an oversampling strategy in DIBR is employed here [19]. Rigid transformation is done after oversampling the depth image in the Kinect V2 with a factor of 10 using the Nearest-Neighbor method.

Evaluation standard: Here, the Root-Mean-Square Error (RMSE) is adopted to evaluate the effects of our proposed method. It estimates the precision in pixels only in the Ultra HD image plane using the same corresponding point pairs as above.

All experiments are conducted on an Intel Core $i3-4030U$ laptop with 16 GB memory and no GPU acceleration.

3.2 Results and Analysis

The quantitative error report of our proposed linear method and the non-linear method is shown in Table I. The linear method outperforms the coarse-to-fine non-linear method. The reason for this may be that the depth information of the points in the ToF sensor of the Kinect V2 is not accurate enough, while the SBA algorithm heavily relies on the accurate structure of these points [20]. The computation time of both algorithms is also exhibited in Table I. The linear method is around 14 times faster than the non-linear one.

The visualization of the final recovered depth image for the Ultra HD camera is illustrated in Fig. 2. Note that, in (a), each pixel corresponds to a depth pixel which is not shown here. It is called registered color image corresponding to a registered depth image, which is plotted with the help of the depth-to-color map in the Kinect for Windows SDK for a better understanding. The recovered color (c) and recovered depth (d) are the recovered results for the Ultra HD camera using the registered color image and registered depth image respectively with our proposed linear method. Both of them have the same resolution as the Ultra HD camera view in (b). It can be found that the depth image (d) is well recovered except for some occlusion regions which are caused by the displacement between cameras.

Both the linear and the non-linear method can be extended to the case of multiple Ultra HD cameras. For lack of space, the case of only two cameras is evaluated here. In addition, the proposed linear calibration method can also be used for recovering the color information for the depth map using a color camera.

4 Conclusion

In this paper, the problem of recovering the depth information of a high resolution Ultra HD camera using a low resolution Kinect V2 sensor is tried to be solved. A linear solution method is proposed for this problem, which is also compared with a coarse-to-fine non-linear method. Experimental results demonstrate the effectiveness and efficiency of this linear method, which performs better than the other. Moreover, the recovered Ultra HD depth image still has room for quality improvement, which will be our next research goal.

Acknowledgement

This project has received funding from the European Union's Framework Programme for Research and Innovation Horizon 2020 (2014-2020) under the Marie Skłodowska-Curie Actions Grant Agreement No. 676401, the Fraunhofer and Max Planck cooperation program within the German pact for research and innovation (PFI), Intel-VCI-CAU and the German Research Foundation (DFG) No. K02044/8-1.

References

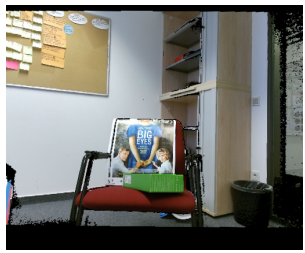
[1] Christoph Fehn, René De La Barré, and Siegmund Pastoor, "Interactive 3-dtv-concepts and key technologies," *Proceedings of the IEEE*, vol. 94, no. 3, pp. 524–538, 2006.

[2] Ting-Chun Wang, Alexei A. Efros, and Ravi Ramamoorthi, "Depth estimation with occlusion modeling using light-field cameras," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 38, no. 11, pp. 2170–2181, 2016.

[3] Łukasz Dąbała, Matthias Ziegler, Piotr Didyk, Frederik Zilly, Joachim Keinert, Karol Myszkowski, H-P Seidel, Przemysław Rokita, and Tobias Ritschel, "Efficient multi-image correspondences for on-line light field video processing," *Computer Graphics Forum*, vol. 35, no. 7, pp. 401–410, 2016.

[4] Beau Tippetts, Dah Jye Lee, Kirt Lillywhite, and James Archibald, "Review of stereo vision algorithms and their suitability for resource-limited systems," *Journal of Real-Time Image Processing*, vol. 11, no. 1, pp. 5–25, 2016.

[5] Xiaoyan Hu and Philippos Mordohai, "A quantitative evaluation of confidence measures for stereo vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, no. 11, pp. 2121–2133, 2012.



(a) Kinect V2 view (registered color image)

(b) Ultra HD camera view

(c) Recovered color

(d) Recovered depth

Figure 2. Experimental results. (a) and (b) are the captured views for the same scene in the Kinect V2 and the Ultra HD cameras. (c) and (d) are the recovered color and depth views in the Ultra HD resolution.

[6] Cha Zhang and Zhengyou Zhang, "Calibration between depth and color sensors for commodity depth cameras," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2011, pp. 1–6.

[7] Daniel Herrera, Juho Kannala, and Janne Heikkilä, "Joint depth and color camera calibration with distortion correction," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 34, no. 10, pp. 2058–2064, 2012.

[8] Miles Hansard, Georgios Evangelidis, Quentin Pelorson, and Radu Horaud, "Cross-calibration of time-of-flight and colour cameras," *Computer Vision and Image Understanding (CVIU)*, vol. 134, pp. 105–115, 2015.

[9] Jiyoung Jung, Joon-Young Lee, Yekeun Jeong, and In So Kweon, "Time-of-flight sensor calibration for a color and depth camera pair," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 37, no. 7, pp. 1501–1513, 2015.

[10] Andrea Corti, Silvio Giancola, Giacomo Mainetti, and Remo Sala, "A metrological characterization of the kinect v2 time-of-flight camera," *Robotics and Autonomous Systems (RAS)*, vol. 75, Part B, pp. 584–594, 2016.

[11] S Zennaro, M Munaro, S Milani, P Zanuttigh, A Bernardi, S Ghidoni, and E Menegatti, "Performance evaluation of the 1st and 2nd generation kinect for multimedia applications," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6.

[12] Vaibhav Vaish, Bennett Wilburn, Neel Joshi, and Marc Levoy, "Using plane + parallax for calibrating dense camera arrays," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004, vol. 1, pp. 2–9.

[13] Richard Hartley and Andrew Zisserman, *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[14] Zhengyou Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 22, no. 11, pp. 1330–1334, 2000.

[15] Radu Horaud, Miles Hansard, Georgios Evangelidis, and Menier Clément, "An overview of depth cameras and range scanners based on time-of-flight technologies," *Machine Vision and Applications (MVA)*, vol. 27, no. 7, pp. 1005–1020, 2016.

[16] Manolis IA Lourakis and Antonis A Argyros, "Sba: A software package for generic sparse bundle adjustment," *ACM Transactions on Mathematical Software (TOMS)*, vol. 36, no. 1, pp. 2, 2009.

[17] Ziyun Cai, Jungong Han, Li Liu, and Ling Shao, "Rgb-d datasets using microsoft kinect or similar sensors: a survey," *Multimedia Tools and Applications (MTA)*, pp. 1–43, 2016.

[18] Valeria Garro, Carlo Dal Mutto, Pietro Zanuttigh, and Guido M Cortelazzo, "A novel interpolation scheme for range data with side information," in *Conference for Visual Media Production (CVMP)*. IEEE, 2009, pp. 52–60.

[19] Sveta Zinger, Luat Do, and PHN de With, "Free-viewpoint depth image based rendering," *Journal of Visual Communication and Image Representation*, vol. 21, no. 5, pp. 533–541, 2010.

[20] Marvin Lindner, Ingo Schiller, Andreas Kolb, and Reinhard Koch, "Time-of-flight sensor calibration for accurate range sensing," *Computer Vision and Image Understanding (CVIU)*, vol. 114, no. 12, pp. 1318–1328, 2010.

6.2 Publication 2

A Novel Kinect v2 Registration Method for Large-Displacement Environments Using Camera and Scene Constraints

Yuan Gao, Sandro Esquivel, Reinhard Koch, Matthias Ziegler, Frederik Zilly and Joachim Keinert

Published in

2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17-20 Sept. 2017, pages 997-1001.

DOI: 10.1109/ICIP.2017.8296431

A NOVEL KINECT V2 REGISTRATION METHOD FOR LARGE-DISPLACEMENT ENVIRONMENTS USING CAMERA AND SCENE CONSTRAINTS

Yuan Gao[†⋆], *Sandro Esquivel*[†], *Reinhard Koch*[†], *Matthias Ziegler*[⋆], *Frederik Zilly*[⋆], *Joachim Keinert*[⋆]

[†] Institute of Computer Science, Christian-Albrechts-University of Kiel, 24118 Kiel, Germany
{yga, sae, rk}@informatik.uni-kiel.de
[⋆] Fraunhofer Institute for Integrated Circuits IIS, 91058 Erlangen, Germany
{matthias.ziegler, frederik.zilly, joachim.keinert}@iis.fraunhofer.de

ABSTRACT

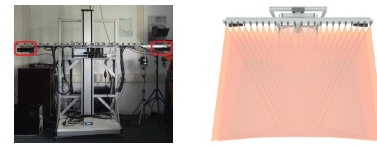
In a lot of multi-Kinect V2-based systems, the registration of these Kinect V2 sensors is an important step which directly affects the system precision. The coarse-to-fine method using calibration objects is an effective way to solve the Kinect V2 registration problem. However, for the registration of Kinect V2 cameras with large displacements, this kind of method may fail. To this end, a novel Kinect V2 registration method, which is also based on the coarse-to-fine framework, is proposed by using camera and scene constraints. Specifically, in the coarse estimation stage, scene constraints are explored using off-the-shelf feature point detectors and camera constraints are explored using homography and fundamental matrices. In the estimation refinement stage, an Iterative Closest Point (ICP)-based point cloud registration method is utilized. Experimental results show that the proposed Kinect V2 registration method using camera and scene constraints performs much better in precision than using calibration objects in the large-displacement environment.

***Index Terms*—** Kinect V2 Registration, Large-Displacement Environment, Coarse-to-Fine Method, Fundamental Matrix, Iterative Closest Point

1. INTRODUCTION

The second version of the Microsoft Kinect (Kinect V2) is one of the most low-cost and high-speed Time-of-Flight (ToF) sensors in the market [1]. The comparison between Kinect V2 and the first generation of Microsoft Kinect (Kinect V1) is well studied in [2, 3, 4, 5], where Kinect V2 exhibits higher accuracy and better performance than Kinect V1 in multimedia applications. A more interesting advantage of Kinect V2 is the possibility of an interference-free multi-Kinect V2 setup. Currently, systems with multiple Kinect V2 sensors have attracted more and more research interests for their wide applications, *e. g.*, people tracking [6, 7], augmented reality [8] and motion capture [9].

Motivation: The multi-camera rig in Fig. 1 is a movable device for capturing dynamic light field built in the Multimedia Information Processing (MIP) laboratory of Kiel University [10]. Two Kinect V2 sensors are integrated in this system for the reason that the Field of View (FOV) of one Kinect V2 is too small compared with the large joint FOV of the other 24 RGB cameras, while utilizing two Kinect V2 can remedy this defect. Accurate registration or calibration of these two Kinect V2 cameras is very tough, considering the distance between these two Kinect V2 cameras is quite large, which is around **2.4** meters. Little literature focuses on this large-displacement depth sensor calibration problem and the traditional checkerboard-based calibration method [11] is prone to fail if the checkerboard is not huge enough for being captured by both cameras at the same time.



(a) Frontal view (b) Virtual vertical view

Fig. 1. The movable multi-camera rig. Red blocks indicate the positions of the Kinect V2 sensors.

Related work: As for multi-Kinect V2 calibration in non-large-displacement setups, several methods have been proposed. Palasek *et al*. leverage a checkerboard to calibrate two Kinect V2 cameras, where up to seven different views of this checkerboard should be captured to improve the calibration precision [12]. Beck *et al*. design an optical tracking system for immersive 3D telepresence which maps points in the depth images into a joint coordinate system with color information using a volumetric calibration and registration approach [13]. However, this approach heavily relies on the tracking of a continuously moving checkerboard. Munaro *et al*. develop a universal framework for scalable people tracking and calibration of different types of depth sensors, including Kinect V2, ToF and stereo cameras [6]. The people detection trajectories are used for calibration refinement. More recently, coarse-to-fine Kinect V2 registration methods are proposed in [14, 15]. In particular, calibration objects, *e. g.*, marker (2D) and wand (1D), are applied in the coarse estimation stage. And Iterative Closest Point [16] and R-Nearest Neighbor [17] approaches are applied in the estimation refinement stage. Nevertheless, both of these two methods rely on specific calibration objects. How to directly make use of camera and scene constraints to register multi-Kinect V2 sensors has not been explored yet.

To solve the Kinect V2 registration problem in the large-displacement environment, a novel coarse-to-fine calibration method using camera and scene constraints is proposed in this paper. To be precise, an off-the-shelf feature detector is utilized to find constraints in the scene, homography and fundamental matrices are employed to construct constraints in the cameras. The coarse estimation is composed of feature point detection, coarse matching, match filtering, and least-squares fitting steps. The estimation refinement consists of an ICP-based point cloud registration algorithm. Experiments are conducted on the movable multi-camera rig with a large displacement between Kinect V2 cameras as introduced above. Experimental results show the validity of the proposed coarse-to-fine registration method using camera and scene constraints in the large-displacement environment.

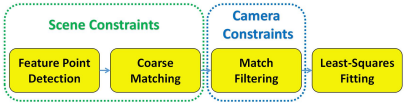


Fig. 2. Flow chart of the coarse estimation step with camera and scene constraints.

2. METHODOLOGY

In this section, coarse-to-fine registration methods, which consist of coarse estimation and estimation refinement steps, are presented after a brief preliminary description to introduce the problem.

2.1. Preliminary

Since there are two Kinect V2 cameras on the multi-camera rig, one Kinect V2 is denoted as C_A, the other one is denoted as C_B for convenience. The capture output of a Kinect V2 is a pair of registered depth and color images. The registered color images are shown in Fig. 3. More details are explained in section 3.1. Therefore, for each Kinect V2 camera, there are two basic coordinate systems. One is camera 3D space, the other is camera image space. The Kinect V2 registration problem can be defined as solving the rigid transformation from C_A 3D space to C_B 3D space, denoted as $\mathbf{R}$ and $\boldsymbol{t}$. The rigid transformation result of the coarse estimation step is denoted as $\mathbf{R}_1$ and $\boldsymbol{t}_1$. The incremental result of the estimation refinement step is denoted as $\mathbf{R}_2$ and $\boldsymbol{t}_2$. Suppose $\boldsymbol{u}_i^a = \begin{bmatrix} u_i^a & v_i^a & 1 \end{bmatrix}^{\mathrm{T}}$ is a point in C_A image space, the corresponding point $\boldsymbol{x}_i^a = \begin{bmatrix} x_i^a & y_i^a & z_i^a & 1 \end{bmatrix}^{\mathrm{T}}$ in the 3D space of C_A can be calculated using the intrinsic camera matrix $\mathbf{K}^a$. Suppose $\boldsymbol{x}_i^a$ and $\boldsymbol{x}_j^b$ correspond to the same point $\boldsymbol{x}_k$ in world 3D space, a good coarse-to-fine calibration should meet this condition:

$$\boldsymbol{x}_j^b = \begin{bmatrix} \mathbf{R}_2 & \boldsymbol{t}_2 \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 & \boldsymbol{t}_1 \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a \tag{1}$$

where:

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_2\mathbf{R}_1 \\ \boldsymbol{t} &= \mathbf{R}_2\boldsymbol{t}_1 + \boldsymbol{t}_2 \end{aligned} \tag{2}$$

Therefore, (2) can be applied for recovering the final rigid transformation using the results of the two stages of coarse-to-fine registration methods.

2.2. Coarse Estimation

Two categories of coarse estimation approaches are introduced in this section. One is based on calibration objects. The other is based on camera and scene constraints.

2.2.1. With Calibration Objects

The calibration object is a tool of making some artificial geometric constraints for the assisted calibration in the coarse estimation stage. The checkerboard is one of the most common calibration tools in computer vision, which is also utilized here. After the corner detection step, n corresponding point pairs in C_A and C_B images spaces are detected. Each pair is expressed as $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ and $\boldsymbol{u}_i^a$ corresponds to a 3D point $\boldsymbol{x}_i^a$ as introduced in section 2.1. The coarse estimation problem can be solved by minimizing the following formula:

$$\begin{gathered} \min_{\mathbf{R}_1, \boldsymbol{t}_1} \sum_{i=1}^{n} \|\boldsymbol{u}_i^b - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^a, \mathbf{K}^b, \mathbf{R}_1, \boldsymbol{t}_1)\|^2 \\ \text{where } \hat{\boldsymbol{u}}(\boldsymbol{x}, \mathbf{K}, \mathbf{R}, \boldsymbol{t}) = \mathrm{proj}(\mathbf{K} \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \end{bmatrix} \boldsymbol{x}) \end{gathered} \tag{3}$$

Here, the Levenberg-Marquardt optimization algorithm is applied to solve this problem.

input : P_A - Point cloud of C_A after the rigid transformation of the coarse estimation stage;
P_B - Point cloud of C_B;
N - Number of point cloud registration iterations;
$\mathbf{R}^a$, $\mathbf{R}^b$ - 3×3 identity matrices;
$\boldsymbol{t}^a$, $\boldsymbol{t}^b$ - 3×1 zero vectors.
output: $\mathbf{R}^a$, $\mathbf{R}^b$, $\boldsymbol{t}^a$, $\boldsymbol{t}^b$.

for $n \leftarrow 1$ **to** N **do**
  $\mathbf{R}, \boldsymbol{t} \leftarrow ICP(P_A, P_B)$;
  for *each point* $\boldsymbol{x}_i^a$ *in* P_A **do**
    $\boldsymbol{x}_i^a \leftarrow \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a$;
  end
  $\mathbf{R}^a \leftarrow \mathbf{R}\mathbf{R}^a$;
  $\boldsymbol{t}^a \leftarrow \mathbf{R}\boldsymbol{t}^a + \boldsymbol{t}$;
  $\mathbf{R}, \boldsymbol{t} \leftarrow ICP(P_B, P_A)$;
  for *each point* $\boldsymbol{x}_i^b$ *in* P_B **do**
    $\boldsymbol{x}_i^b \leftarrow \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^b$;
  end
  $\mathbf{R}^b \leftarrow \mathbf{R}\mathbf{R}^b$;
  $\boldsymbol{t}^b \leftarrow \mathbf{R}\boldsymbol{t}^b + \boldsymbol{t}$;
end

Algorithm 1: Point cloud registration

2.2.2. With Camera and Scene Constraints

There are four basic steps in the coarse estimation stage using camera and scene constraints as illustrated in Fig. 2. Scene constraints are calculated in the first two steps. Camera constraints help reject outliers in the third step. Details of these four steps are explained as below.

Feature point detection: Interest point detection is a well studied research area in computer vision [18]. Speeded Up Robust Features (SURF) [19] is used here for the reason that it is a fast and robust feature descriptor which is suitable for the feature detection task of this work. Besides, there are some holes in the registered color images of Kinect V2 sensors, which result in the bad performance of some other interest point detectors. Comparisons of different point detection methods can be done in the Full High Definition (FHD) resolution images of the Kinect V2 RGB sensor, while the exact transformation relationship between RGB and ToF sensors in a Kinect V2 is unknown but studied in [20, 21, 22], which is beyond the research scope of this paper.

Coarse matching: The k-Nearest-Neighbors (KNN) [23] and the ratio test [24] are utilized to match the detected feature points from the image spaces of two cameras.

Match filtering: In this step, camera constraints are utilized to filter the outlier matches with the RANdom SAmple Consensus (RANSAC) [25] framework. Camera constraints include homography and fundamental matrices [26] described as:

$$\begin{aligned} \boldsymbol{u}_i^b &\sim \mathbf{H}\boldsymbol{u}_i^a \\ (\boldsymbol{u}_i^b)^{\mathrm{T}}\mathbf{F}\boldsymbol{u}_i^a &= 0 \end{aligned} \tag{4}$$

Here, $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ is a corresponding point pair in the camera image planes of C_A and C_B after the coarse matching process. The RANSAC algorithm is employed to calculate $\mathbf{H}$ and $\mathbf{F}$ robustly, which are then used to keep the inlier matches.

Least-squares fitting: The inlier matches $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ are transformed to $(\boldsymbol{x}_i^a, \boldsymbol{x}_i^b)$ in the camera 3D spaces of C_A and C_B using

Table I. RMSE of the coarse-to-fine registration methods.

Coarse Estimation	RMSE (mm)	Estimation Refinement	RMSE (mm)
Checkerboard-based	**78.33**	ICP-based point cloud registration	84.11
Homography matrix-based	302.05	ICP-based point cloud registration	44.82
Fundamental matrix-based	295.58	ICP-based point cloud registration	**34.34**

$\mathbf{K}^a, \mathbf{K}^b$ and the depth information from the registered depth image. The least-squares fitting algorithm [27] is the process of minimizing the distance between two 3D point sets using the Singular Value Decomposition (SVD) method:

$$\min_{\mathbf{R}_1, \boldsymbol{t}_1} \sum_{i=1}^{n} \| \boldsymbol{x}_i^b - \begin{bmatrix} \mathbf{R}_1 & \boldsymbol{t}_1 \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a \|^2 \tag{5}$$

2.3. Estimation Refinement

The Iterative Closest Point (ICP) algorithm is an effective method for the registration of two similar point clouds without explicitly given correspondences [28]. The reason why ICP is chosen here is that it has similar performance in this experimental setup compared with other methods, *e. g.*, R-Nearest Neighbor [17], 3D feature detection and matching [29], which is also stated in [15]. A common point cloud registration algorithm for point clouds of two cameras is illustrated in Algorithm 1. Note that P_A is a transformed point cloud of C_A after the coarse estimation stage. Suppose $\boldsymbol{x}_i^a$ is a point in P_A, which corresponds to a point $\boldsymbol{x}_j^b$ in P_B. Both of them also correspond to the same point $\boldsymbol{x}_k$ in world 3D space. After using the point cloud registration algorithm, the following formula should hold ideally:

$$\boldsymbol{x}_k = \begin{bmatrix} \mathbf{R}^a & \boldsymbol{t}^a \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a = \begin{bmatrix} \mathbf{R}^b & \boldsymbol{t}^b \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_j^b \tag{6}$$

Therefore, the rigid transformation of estimation refinement step is:

$$\begin{aligned} \mathbf{R}_2 &= (\mathbf{R}^b)^{\mathrm{T}} \mathbf{R}^a \\ \boldsymbol{t}_2 &= (\mathbf{R}^b)^{\mathrm{T}} (\boldsymbol{t}^a - \boldsymbol{t}^b) \end{aligned} \tag{7}$$

(a) C_A view of the checkerboard (b) C_B view of the checkerboard

(c) C_A view of the scene (d) C_B view of the scene

Fig. 3. Registered color images for experiments.

3. EXPERIMENTS

3.1. Experimental Settings

Experiments are implemented on the movable multi-camera rig as introduced in section 1. Two Kinect V2 cameras C_A and C_B are connected to only one control computer using the open-source Kinect V2 driver, *libfreenect2*[1]. The capture process of both cameras is synchronized internally with signal and time stamp technologies. Using this Kinect V2 driver, the intrinsic camera matrices $\mathbf{K}^a$ and $\mathbf{K}^b$ can be accessed. Besides, the captured views of Kinect V2 sensors are pairs of registered color and depth images with lens distortion corrected. An example output registered color image is shown in Fig. 3 (a). Note that each color pixel in this image has a corresponding depth value in the paired registered depth image, and pixels with color information missing mean that depth information for these pixels can not be accessed from the sensors. The resolutions of both registered color and depth images are 512×424 pixels.

Checkerboard-captured data: A checkerboard is put in front of the multi-camera rig at a distance of around 2.8 m. This checkerboard has 28 (4×7) inner corners and the size of each square in it is 124×124 mm. Registered color images for the views of this checkerboard in C_A and C_B are illustrated in Fig. 3 (a)(b). This data is used for the coarse estimation step with a calibration object as described in section 2.2.1 and the evaluation metric as described below.

Scene-captured data: A natural scene of a conference room of the size of $5.5 \times 3.0 \times 7.8$ m ($w \times h \times d$) is captured without artificial calibration objects in it. Registered color images for the views of this scene in C_A and C_B are illustrated in Fig. 3 (c)(d). This data is used for the coarse estimation with camera and scene constraints from section 2.2.2 and the estimation refinement step from section 2.3.

Coarse estimation details: Automatic corner detection and SURF detection are implemented with OpenCV. Parameter r for ratio test is set to 0.6. The threshold ε in RANSAC is set to 1.0 pixels.

Estimation refinement details: The number of point cloud registration iterations N is set to 10. In each point cloud registration iteration, the ICP function has 5 iterations.

Evaluation metric: The Root-Mean-Square Error (RMSE) is adopted to evaluate the effects of coarse-to-fine registration methods with the checkerboard-captured data. Using the same corresponding point pair definition $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ as described in section 2.2.1, the number of corresponding point pairs n is equal to 28. The RMSE is defined as:

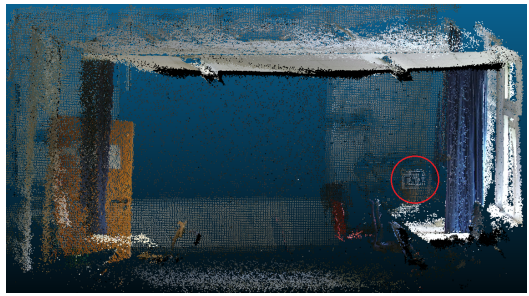
$$\sqrt{\frac{1}{n} \sum_{i=1}^{n} \| \boldsymbol{x}_i^b - \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \boldsymbol{x}_i^a \|^2} \tag{8}$$

Comparison experiments are conducted on an Intel Core i3 − 4030U laptop with 16 GB RAM and no GPU acceleration, using the captured datasets from the control computer. Both source code and datasets are going to be released on our website[2].

3.2. Results and Analysis

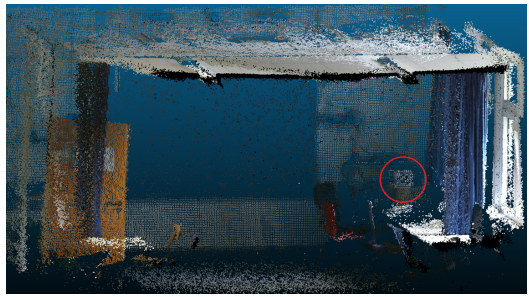
Quantitative evaluation results of different coarse-to-fine registration methods are exhibited in Table I. In the coarse estimation phase,

[1] http://dx.doi.org/10.5281/zenodo.50641
[2] https://ygaokiel.github.io



(a) Checkerboard-based coarse estimation with estimation refinement

(b)



(c) Fundamental matrix-based coarse estimation with estimation refinement

(d)

Fig. 4. Results of registered point clouds with coarse-to-fine registration methods. Red circle areas of (a) and (c) are amplified in (b) and (d).

the checkerboard-based method achieves much more precise results than the homography and fundamental matrices-based methods, which shows that the calibration object is an effective assistance tool to improve the initial calibration. Then, in the estimation refinement phase, the precision of the checkerboard-based method decreases a little bit, while the precision of coarse estimation methods using camera and scene constraints improves dramatically. Specifically, fundamental matrix-based coarse estimation plus estimation refinement has the best performance among these three coarse-to-fine registration methods. The reason for this may be that the coarse estimation with camera and scene constraints gives a globally optimal start point for the ICP-based point cloud registration method, while the coarse estimation using the calibration object offers a locally optimal start point, which leads to its failure in this large-displacement setup. Also, ICP remedies wrong matches that might still remain after the filtering in the coarse estimation step.

Qualitative evaluation results are also presented as shown in Fig. 4 using the scene-captured data. Red circles in Fig. 4 (a)(c) indicate the same white box on the table. The result of checkerboard-based coarse estimation with estimation refinement in Fig. 4 (a) has obvious non-overlapping parts in the red circle area. However, in Fig. 4 (c), the two point clouds coincide very well in the location marked with the red circle, which indicates the effectiveness of coarse estimation with camera and scene constraints again.

4. CONCLUSION

In this paper, camera and scene constraints are exploited inside a coarse-to-fine framework to solve the Kinect V2 registration problem in the large-displacement environment. The proposed Kinect V2 registration method uses homography and fundamental matrix estimations from 2D correspondences found with a SURF detector to estimate the camera pose. The fundamental matrix-based coarse-to-fine registration method outperforms the checkerboard-based coarse-to-fine registration method on a multi-camera rig with a large displacement between two Kinect V2 sensors, which proves the effectiveness of the proposed Kinect V2 registration method for large-displacement environments. How to exploit the camera and scene constraints in the FHD-resolution RGB sensors of Kinect V2 devices to solve the Kinect V2 registration problem in the large-displacement environment will be our next research goal.

5. ACKNOWLEDGMENTS

The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, Intel-VCI-CAU, the German Research Foundation (DFG) No. K02044/8-1 and the Fraunhofer and Max Planck cooperation program within the German pact for research and innovation (PFI).

6. REFERENCES

[1] Andrea Corti, Silvio Giancola, Giacomo Mainetti, and Remo Sala, "A metrological characterization of the Kinect v2 Time-of-Flight camera," *Robotics and Autonomous Systems (RAS)*, vol. 75, pp. 584–594, 2016.

[2] Oliver Wasenmüller and Didier Stricker, "Comparison of Kinect v1 and v2 depth images in terms of accuracy and precision," in *Asian Conference on Computer Vision Workshops (ACCVW)*, 2016.

[3] Marek Kraft, Michał Nowicki, Adam Schmidt, Michał Fularz, and Piotr Skrzypczyński, "Toward evaluation of visual navigation algorithms on RGB-D data from the first- and second-generation Kinect," *Machine Vision and Applications (MVA)*, pp. 1–14, 2016.

[4] Hamed Sarbolandi, Damien Lefloch, and Andreas Kolb, "Kinect range sensing: Structured-light versus Time-of-Flight Kinect," *Computer Vision and Image Understanding (CVIU)*, vol. 139, pp. 1–20, 2015.

[5] S Zennaro, M Munaro, S Milani, P Zanuttigh, A Bernardi, S Ghidoni, and E Menegatti, "Performance evaluation of the 1st and 2nd generation Kinect for multimedia applications," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6.

[6] Matteo Munaro, Filippo Basso, and Emanuele Menegatti, "Openptrack: Open source multi-camera calibration and people tracking for RGB-D camera networks," *Robotics and Autonomous Systems (RAS)*, vol. 75, pp. 525–538, 2016.

[7] Can Wang, Hong Liu, and Yuan Gao, "Scene-adaptive hierarchical data association for multiple objects tracking," *IEEE Signal Processing Letters (SPL)*, vol. 21, no. 6, pp. 697–701, 2014.

[8] Andrea Canessa, Manuela Chessa, Agostino Gibaldi, Silvio P Sabatini, and Fabio Solari, "Calibrated depth and color cameras for accurate 3D interaction in a stereoscopic augmented reality environment," *Journal of Visual Communication and Image Representation*, vol. 25, no. 1, pp. 227–237, 2014.

[9] Licong Zhang, Jürgen Sturm, Daniel Cremers, and Dongheui Lee, "Real-time human motion tracking using multiple depth cameras," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 2389–2395.

[10] Sandro Esquivel, Yuan Gao, Tim Michels, Luca Palmieri, and Reinhard Koch, "Synchronized data capture and calibration of a large-field-of-view moving multi-camera light field rig," in *3DTV-CON Workshops*, 2016.

[11] Zhengyou Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 22, no. 11, pp. 1330–1334, 2000.

[12] Petar Palasek, Heng Yang, Zongyi Xu, Navid Hajimirza, E-broul Izquierdo, and Ioannis Patras, "A flexible calibration method of multiple Kinects for 3D human reconstruction," in *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 2015, pp. 1–4.

[13] Stephan Beck and Bernd Froehlich, "Volumetric calibration and registration of multiple RGBD-sensors into a joint coordinate system," in *IEEE Symposium on 3D User Interfaces (3DUI)*, 2015, pp. 89–96.

[14] Marek Kowalski, Jacek Naruniec, and Michal Daniluk, "LiveScan3D: A fast and inexpensive 3D data acquisition system for multiple Kinect v2 sensors," in *IEEE International Conference on 3D Vision (3DV)*, 2015, pp. 318–325.

[15] Diana-Margarita Córdova-Esparza, Juan R Terven, Hugo Jiménez-Hernández, and Ana-Marcela Herrera-Navarro, "A multiple camera calibration and point cloud fusion tool for Kinect v2," *Science of Computer Programming (SCP)*, 2016.

[16] Paul J Besl and Neil D McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 14, no. 2, pp. 239–256, 1992.

[17] Alexandr Andoni, *Nearest neighbor search: the old, the new, and the impossible*, Ph.D. thesis, Massachusetts Institute of Technology, 2009.

[18] Steffen Gauglitz, Tobias Höllerer, and Matthew Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International Journal of Computer Vision (IJCV)*, vol. 94, no. 3, pp. 335–360, 2011.

[19] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "SURF: Speeded up robust features," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 404–417.

[20] Yuan Gao, Matthias Ziegler, Frederik Zilly, Sandro Esquivel, and Reinhard Koch, "A linear method for recovering the depth of Ultra HD cameras using a Kinect v2 sensor," in *IAPR International Conference on Machine Vision Applications (MVA)*, 2017, pp. 464–467.

[21] Wanbin Song, Anh Vu Le, Seokmin Yun, Seung-Won Jung, and Chee Sun Won, "Depth completion for Kinect v2 sensor," *Multimedia Tools and Applications (MTA)*, pp. 1–24, 2016.

[22] Oliver Wasenm, Marcel Meyer, and Didier Stricker, "Corbs: Comprehensive RGB-D benchmark for slam using Kinect v2," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1–7.

[23] Naomi S Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[24] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.

[25] Martin A Fischler and Robert C Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] Richard Hartley and Andrew Zisserman, *Multiple view geometry in computer vision*, Cambridge university press, 2003.

[27] K. Somani Arun, Thomas S. Huang, and Steven D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. PAMI-9, no. 5, pp. 698–700, 1987.

[28] François Pomerleau, Francis Colas, Roland Siegwart, and Stéphane Magnenat, "Comparing ICP variants on real-world data sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, 2013.

[29] Tal Darom and Yosi Keller, "Scale-invariant features for 3-D mesh models," *IEEE Transactions on Image Processing (TIP)*, vol. 21, no. 5, pp. 2758–2769, 2012.

6.3 Publication 3

A Novel Self-Calibration Method for a Stereo-ToF System Using a Kinect V2 and Two 4K GoPro Cameras

Yuan Gao, Sandro Esquivel, Reinhard Koch and Joachim Keinert

Published in

2017 International Conference on 3D Vision (3DV), Qingdao, China, 10-12 Oct. 2017, pages 21-28.

DOI: 10.1109/3DV.2017.00013

A Novel Self-Calibration Method for a Stereo-ToF System Using a Kinect V2 and Two 4K GoPro Cameras

Yuan Gao, Sandro Esquivel, Reinhard Koch
Christian-Albrechts-University of Kiel
24118 Kiel, Germany
{yga, sae, rk}@informatik.uni-kiel.de

Joachim Keinert
Fraunhofer Institute for Integrated Circuits IIS
91058 Erlangen, Germany
joachim.keinert@iis.fraunhofer.de

Abstract

A new light-field movie capture device using a Kinect V2 sensor and two 4K GoPro cameras is presented in this paper. Due to the uncontrollable tilt of the Kinect V2 camera, it is hard to obtain a constant rigid transformation between the stereo- and ToF-camera systems. To this end, a novel self-calibration method is proposed, which takes advantage of the geometric constraints from the scene and the cameras. Specifically, a camera orientation approximation approach is utilized to estimate the rigid transformation of the stereo-ToF system based on reliable point pairs filtered by the geometric constraints. Besides, a depth correction step is exploited to improve the depth accuracy of the Kinect V2 sensor. Moreover, a depth fusion strategy for the stereo- and ToF-depth data is proposed to provide more accurate depth images in 4K resolution. Experimental results demonstrate the effectiveness of the proposed depth correction step, stereo-ToF calibration method and depth fusion strategy.

1. Introduction

With more and more 4K Ultra High Definition (UHD) 3D TVs and high-fidelity Virtual Reality (VR) Head-Mounted Displays (HMDs) having been launched into the consumer market, how to create high-resolution and high-quality contents for these devices is becoming a research hotspot. The Depth-Image-Based Rendering (DIBR) approach is such kind of method which is capable of producing free-viewpoint videos for the above devices [33, 42, 9]. Since the accuracy and the resolution of the depth images in DIBR are the critical influence factors for the rendering of free-viewpoint videos, in this paper, how to recover the depth information for a 4K resolution RGB camera is investigated. Currently, the recovery of the depth information using multiple RGB cameras has been well researched, which can be classified into two categories. One is the light field-to-depth approach [35, 4], the other is the stereo matching method [34, 19]. Apart from the RGB cameras-based algorithms, the depth information can also be acquired from external sensors. For example, the second version of the Microsoft Kinect (Kinect V2) is one of the most low-cost and high-speed Time-of-Flight (ToF) sensors in the market [3]. The comparison between the Kinect V2 and the first generation of Microsoft Kinect (Kinect V1) is well studied in [36, 21, 31, 38], where the Kinect V2 exhibits a better performance in a lot of multimedia applications than the Kinect V1.

(a) Frontal view (b) Vertical view

Figure 1. A multi-camera rig for capturing light-field movies. The Kinect V2 sensor is in the middle of the two 4K GoPro cameras.

Motivation: The multi-camera rig in Fig. 1 is a prototypical movie capture system for making light-field movies. Two GoPro Hero3+ cameras with a Kinect V2 sensor are mounted on this rig. In order to increase angular resolution and minimize lens distortion, the original lenses of both GoPro cameras have been replaced by two customized lenses with the same Field of View (FOV) of about 70 degrees [4]. The camera resolutions of these two GoPro cameras are 4K $(4,000 \times 3,000)$. There are two challenging problems of this multi-camera rig that need to be solved. First, the GoPro cameras are well fixed on the camera rig, while the Kinect V2 is impossible to be fixed because of the changeable tilt of it, which easily leads to the inconsistency of the camera extrinsic parameters after the multi-camera rig being moved. How to automatically calibrate this stereo-ToF system using the scene and camera constraints is challenging. The other challenging problem is that it is difficult to recover a 4K-resolution depth image for one of the GoPro

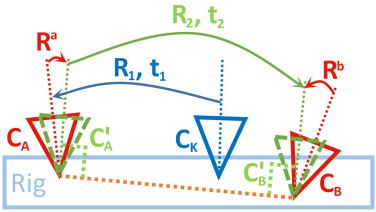


Figure 2. A virtual vertical view of the multi-camera rig for describing the camera calibration problems.

cameras, considering that the distance between the two GoPro cameras is quite large (around 39 cm), and little literature focuses on the stereo- and ToF-depth fusion problem in 4K resolution.

To solve the two challenging problems of the multi-camera rig, a depth correction step, a stereo-ToF calibration method and a depth fusion strategy are proposed in this paper. In particular, the depth correction step increases the depth accuracy of the Kinect V2 camera. The stereo-ToF calibration method is based on the reliable point pairs, which are detected by an off-the-shelf feature point detector and filtered using geometric constraints in the cameras and the scene [12]. Besides, the camera rotation matrix can be linearly approximated because the Kinect V2 and the GoPro cameras have similar orientations. The depth fusion strategy exploits the rigid transformation result of the stereo-ToF calibration method to fuse the depth information from the stereo matching method and the ToF sensor at the pixel level. Experimental results show that the depth correction step contributes to the stereo-ToF calibration. The stereo-ToF calibration method using the camera orientation approximation achieves the best performance compared with baseline approaches. Moreover, the depth fusion strategy is capable of creating depth images of better quality than using the stereo matching method or the ToF sensor alone.

The paper is organized as follows. Section 2 focuses on the introduction of related works. Section 3 outlines a depth correction step, a coarse-to-fine camera calibration framework using the reliable point pairs from the filtering of the scene and camera constraints, and a pixel-level depth fusion strategy. Section 4 is devoted to the experiments and analysis of the camera calibration and the depth fusion. Finally, section 5 concludes and summarizes this paper.

2. Related Work

The Perspective-n-Point (PnP) problem is first described in [10], which stands for the problem of how to estimate the camera pose of a calibrated camera using n known 3D reference points in the world coordinate frame and their corresponding 2D points on the camera image plane of this calibrated camera. The solutions to the PnP problem can be classified into two categories: iterative and non-iterative methods. As for the iterative-based methods, Lu *et al*. minimize an object-space collinearity error for computing orthogonal rotation matrices, which is proven to be globally convergent [28]. Zhang proposes a closed-form solution for estimating the camera intrinsic and extrinsic parameters, and then refines them with the Levenberg-Marquardt algorithm [39]. With regard to the non-iterative-based methods, Lepetit *et al*. express the non-iterative solution to the PnP problem as a vector standing for a weighted sum of the null eigenvectors and their method achieves the computational complexity in n [22]. Li *et al*. also present an O(n) solution by estimating the coordinates of two special end points [23].

As for the fusion of stereo- and ToF-depth data in non-4K resolution, several methods have been proposed [30]. Zhu *et al*. use the MAP-MRF Bayesian framework to solve the stereo and ToF data fusion problem and a belief propagation-based method is applied to fulfill the depth inference [41, 40]. Gandhi *et al*. utilize a Bayesian fusion method within an efficient seed-growing algorithm to solve the same problem [11]. More recently, Dal Mutto *et al*. also use the MAP-MRF framework to fuse the stereo- and ToF-depth data with considering the mixed pixel effect [6]. Evangelidis *et al*. address the stereo-ToF fusion problem by solving a set of local energy optimization problems hierarchically [8]. Marin *et al*. extend the Local Consistency (LC) fusion framework of [7] with taking into account of the depth data confidence [29]. With regard to the calibration between the stereoscopic camera pair and the ToF sensor in the publicly-available datasets [8, 6], checkerboard-based methods are taken advantage of [16, 5]. However, both of these two datasets do not contain color image contents in 4K resolution.

3. Methodology

In this section, a coarse-to-fine framework for the calibration process of the stereo-ToF system is presented after a brief introduction to the self-calibration problem and the reliable point pair detection step. Besides, a depth fusion strategy for different depth sources is described in the end of this section.

3.1. Preliminary

Since there are two GoPro cameras and one Kinect V2 sensor on the multi-camera rig, for the sake of describing convenience, the left and right GoPro cameras are denoted as $\mathbb{C}_\mathrm{A}$ and $\mathbb{C}_\mathrm{B}$ respectively, and the Kinect V2 camera is denoted as $\mathbb{C}_\mathrm{K}$. Each camera has two basic spaces: camera 3D space and camera image space. The intrinsic matrices of the GoPro and Kinect V2 cameras are defined as $\mathbf{K}_a$, $\mathbf{K}_b$, $\mathbf{K}_k$ respectively, and the lens distortions of them are assumed to

have been corrected. It should be noted that the coordinate system of the Kinect V2 camera $\mathbb{C}_\mathrm{K}$ coincides with that of the RGB sensor in it, by which color and depth images in Full High Definition (FHD) resolution are captured. More details concerning this are explained in section 4.1.

The self-calibration of this stereo-ToF system is defined as to estimate the rigid transformation ($\mathbf{R}_1$, $\boldsymbol{t}_1$) from the Kinect V2 sensor $\mathbb{C}_\mathrm{K}$ to the left GoPro camera $\mathbb{C}_\mathrm{A}$. The calibration of the stereoscopic GoPro camera pair is expressed as to measure the rotation rectification matrices $\mathbf{R}^a$ and $\mathbf{R}^b$, which turn $\mathbb{C}_\mathrm{A}$ into a virtual camera $\mathbb{C}'_\mathrm{A}$ and $\mathbb{C}_\mathrm{B}$ into another virtual camera $\mathbb{C}'_\mathrm{B}$. Typically, the intrinsic camera matrices of the virtual cameras $\mathbb{C}'_\mathrm{A}$ and $\mathbb{C}'_\mathrm{B}$ are the same, *i.e.* $\mathbf{K}'_a = \mathbf{K}'_b$. The straight line going through the optical centers of both GoPro cameras are parallel to the coplanar camera image planes of $\mathbb{C}'_\mathrm{A}$ and $\mathbb{C}'_\mathrm{B}$. The rigid transformation from $\mathbb{C}'_\mathrm{A}$ to $\mathbb{C}'_\mathrm{B}$ is then defined as ($\mathbf{R}_2$, $\boldsymbol{t}_2$), where $\mathbf{R}_2$ should be an identity matrix. The above calibration descriptions are illustrated in Fig. 2 as well. It can be found that $\mathbb{C}_\mathrm{K}$ is not in the middle of the stereo GoPro pair. The reason is that the RGB sensor in $\mathbb{C}_\mathrm{K}$ is physically closer to $\mathbb{C}_\mathrm{B}$ than to $\mathbb{C}_\mathrm{A}$ as shown in Fig. 1 (a).

Depth Correction: The depth accuracy of a Kinect V2 camera has a constant offset of -18 mm, which is well evaluated in [36]. It is important to correct the depth images from the Kinect V2 sensor by compensating this accuracy offset before fusing the stereo- and ToF-depth data. Otherwise the misalignment of the point clouds derived from the ToF sensor and the stereo matching method would happen, which is analyzed in section 4.2.

3.2. Reliable Point Pair Detection

Interest point detection has been well studied in the computer vision field [15]. The traditional SIFT keypoint detector and descriptor are used here for their robustness [27]. Another reason for only evaluating SIFT is that, since $\mathbb{C}_\mathrm{K}$ is able to capture FHD resolution color images and $\mathbb{C}_\mathrm{A}$ can capture 4K resolution images, any classical interest point detection algorithm is capable of detecting adequate reliable feature points for the following processes. Besides, the influence of choosing another type of keypoint detector and descriptor on the calibration result is negligible for the experiments.

The detected feature points in the camera image spaces of $\mathbb{C}_\mathrm{A}$ and $\mathbb{C}_\mathrm{K}$ are then exploited to compose reliable matched point pairs. Here, the k-Nearest-Neighbors (KNN) [1] and ratio test [27] methods are utilized to fulfill this task. Afterwards, the resulting corresponding pairs still contain some outliers, which are filtered by using epipolar constraints with the RANdom SAmple Consensus (RANSAC) algorithm [10, 17]. The remaining corresponding point pairs are assumed to be accurate for the subsequent processes.

3.3. Coarse-to-Fine Framework

A coarse-to-fine framework [13] is generally composed of a coarse estimation step based on the solution to the PnP problem, and an estimation refinement step based on the bundle adjustment algorithm [25], which are introduced as follows.

3.3.1 Coarse Estimation

Suppose the number of the corresponding point pairs is n. A corresponding point pair is denoted as ($\boldsymbol{u}_i^a$, $\boldsymbol{u}_i^k$), where $\boldsymbol{u}_i^a = \begin{bmatrix} u_i^a & v_i^a & 1 \end{bmatrix}^\mathrm{T}$ is a 2D point in the camera image space of $\mathbb{C}_\mathrm{A}$, and $\boldsymbol{u}_i^k$ is a 2D point in the camera image space of $\mathbb{C}_\mathrm{K}$, having the same format as $\boldsymbol{u}_i^a$. For $\boldsymbol{u}_i^k$, the corresponding 3D point $\boldsymbol{x}_i^k = \begin{bmatrix} x_i^k & y_i^k & z_i^k & 1 \end{bmatrix}^\mathrm{T}$ in the camera 3D space of $\mathbb{C}_\mathrm{K}$ is calculated by using the intrinsic camera matrix $\mathbf{K}_k$ and the depth information of $\mathbb{C}_\mathrm{K}$. The coarse estimation step is essentially to estimate the rigid transformation from the camera coordinates of $\mathbb{C}_\mathrm{K}$ to the camera coordinates of $\mathbb{C}_\mathrm{A}$ by calculating the below formula:

$$\min_{\mathbf{R}_1^2, \boldsymbol{t}_1^2} \sum_{i=1}^{n} \|\boldsymbol{u}_i^a - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^k, \mathbf{K}_a, \mathbf{R}_1^2, \boldsymbol{t}_1^2)\|^2, \tag{1}$$

where:

$$\hat{\boldsymbol{u}}(\boldsymbol{x}, \mathbf{K}, \mathbf{R}, \boldsymbol{t}) = \mathrm{proj}(\mathbf{K} \begin{bmatrix} \mathbf{R} & \boldsymbol{t} \end{bmatrix} \boldsymbol{x}). \tag{2}$$

The results of the coarse estimation stage are denoted as ($\mathbf{R}_1^2$, $\boldsymbol{t}_1^2$). Several methods have been proposed for solving the above PnP problem, *e.g.*, LHM [28], EPnP [22], and RPnP [23]. Based on the specific camera structure of our stereo-ToF system, a camera orientation approximation-based PnP solution is presented as below.

Camera Orientation Approximation: When observing the camera configuration in Fig. 1, it can be found that $\mathbb{C}_\mathrm{K}$ and $\mathbb{C}_\mathrm{A}$ have a minor orientation difference, which indicates that the rotation matrix $\mathbf{R}_1^2$ can be approximated by a linear method in [26]. Suppose the camera orientation difference between $\mathbb{C}_\mathrm{K}$ and $\mathbb{C}_\mathrm{A}$ is expressed as $\boldsymbol{r} = \begin{bmatrix} \alpha & \beta & \gamma \end{bmatrix}^\mathrm{T}$. The approximated rotation matrix $\mathbf{R}_1^2$ is denoted as:

$$\mathbf{R}_1^2 = \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}. \tag{3}$$

After transforming the 2D image point $\boldsymbol{u}_i^a$ into a 2D point $\boldsymbol{p}_i^a = \begin{bmatrix} p_i^a & q_i^a & 1 \end{bmatrix}^\mathrm{T}$ on the normalized image plane of $\mathbb{C}_\mathrm{A}$ with using $\mathbf{K}_a$, the rigid transformation progress can be written as:

$$\begin{bmatrix} \mathbf{R}_1^2 & \boldsymbol{t}_1^2 \end{bmatrix} \boldsymbol{x}_i^k = \lambda \boldsymbol{p}_i^a. \tag{4}$$

The scaling factor λ can be derived from the combination of equation (3) and (4), expressed as follows:

$$\lambda = \begin{bmatrix} -\beta & \alpha & 1 & \boldsymbol{t}_1^2(3) \end{bmatrix} \boldsymbol{x}_i^k. \tag{5}$$

Afterwards, equation (4) is written as:

$$\mathbf{A}\begin{bmatrix}\boldsymbol{r}\\ \boldsymbol{t}_1^2\end{bmatrix} = \begin{bmatrix}p_i^a z_i^k - x_i^k\\ q_i^a z_i^k - y_i^k\end{bmatrix}, \tag{6}$$

where:

$$\mathbf{A} = \begin{bmatrix}-p_i^a y_i^k & (p_i^a x_i^k + z_i^k) & -y_i^k & 1 & 0 & -p_i^a\\ -(q_i^a y_i^k + z_i^k) & q_i^a x_i^k & x_i^k & 0 & 1 & -q_i^a\end{bmatrix}. \tag{7}$$

The linear least-squares problem presented in equation (6) and (7) can be solved by using the SVD algorithm to compute a pseudo-inverse or using normal equations, requiring at least three corresponding point pairs, *i.e.* $n \geqslant 3$. The approximated rotation matrix $\mathbf{R}_1^2$ is then converted to a standard rotation matrix by normalization.

3.3.2 Estimation Refinement

Due to the depth precision and flying pixel problems of any Kinect V2 device [36, 31, 24], the 3D point $\boldsymbol{x}_i^k$ generated from $\boldsymbol{u}_i^k$ is not equal to the ground truth 3D point in the camera coordinate system of $\mathbb{C}_\mathrm{K}$, which is further refined by using the formula as below:

$$\min_{\mathbf{R}_1^j, \boldsymbol{t}_1^j, \boldsymbol{x}_i^k} \sum_{j=1}^{2}\sum_{i=1}^{n} \|\boldsymbol{u}_i^j - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^k, \mathbf{K}_j, \mathbf{R}_1^j, \boldsymbol{t}_1^j)\|^2. \tag{8}$$

Here, the input parameter $\mathbf{R}_1^1$ is a 3×3 identity matrix and $\boldsymbol{t}_1^1$ is a zero vector. Besides, the input parameters $\mathbf{R}_1^2$ and $\boldsymbol{t}_1^2$ are the results of the previous coarse estimation step. When $j = 1$, it indicates that j-relevant parameters are also related to camera $\mathbb{C}_\mathrm{K}$. Therefore, $\mathbf{K}_1$ is as same as $\mathbf{K}_k$, and $\boldsymbol{u}_i^1$ is equal to $\boldsymbol{u}_i^k$. As for $j = 2$, it turns into the case of camera $\mathbb{C}_\mathrm{A}$ where $\mathbf{K}_2 = \mathbf{K}_a$ and $\boldsymbol{u}_i^2 = \boldsymbol{u}_i^a$. The nonlinear least-squares optimization problem defined in (8) can be solved by a robust bundle adjustment approach efficiently [37].

The final rigid transformation ($\mathbf{R}_1$, $\boldsymbol{t}_1$) from the camera coordinates of $\mathbb{C}_\mathrm{K}$ to the camera coordinates of $\mathbb{C}_\mathrm{A}$ is expressed as follows:

$$\begin{aligned}\mathbf{R}_1 &= \mathbf{R}_1^2(\mathbf{R}_1^1)^\mathrm{T},\\ \boldsymbol{t}_1 &= \boldsymbol{t}_1^2 - \mathbf{R}_1\boldsymbol{t}_1^1.\end{aligned} \tag{9}$$

Note that, equation (9) can also stand for the stereo-ToF calibration result of only using the coarse estimation method by replacing $\mathbf{R}_1^1$ with an identity matrix, and $\boldsymbol{t}_1^1$ with a zero vector.

3.4. Depth Fusion

The depth data of the Kinect V2 sensor and the stereo matching method are fused together in the camera image space of $\mathbb{C}'_\mathrm{A}$. The rigid transformation ($\mathbf{R}_3$, $\boldsymbol{t}_3$) from the camera coordinates of $\mathbb{C}_\mathrm{K}$ to the camera coordinates of $\mathbb{C}'_\mathrm{A}$ is denoted as:

$$\begin{aligned}\mathbf{R}_3 &= \mathbf{R}^a\mathbf{R}_1,\\ \boldsymbol{t}_3 &= \mathbf{R}^a\boldsymbol{t}_1.\end{aligned} \tag{10}$$

The 3D points in the camera 3D space of $\mathbb{C}_\mathrm{K}$ are projected onto the camera image plane of $\mathbb{C}'_\mathrm{A}$ using ($\mathbf{R}_3$, $\boldsymbol{t}_3$) and $\mathbf{K}'_a$. Since there exists an image resolution difference between $\mathbb{C}_\mathrm{K}$ and $\mathbb{C}'_\mathrm{A}$, the above 3D-point-projection solution may cause information loss, *i.e.* sparse points on the destination image plane. To solve this problem, a universal oversampling strategy in DIBR is employed here [42]. The oversampling rate s (= 4) is applied to adjust the resolution of all the captured images of $\mathbb{C}_\mathrm{K}$ and the intrinsic camera matrix $\mathbf{K}_k$. Afterwards, in the camera image space of $\mathbb{C}'_\mathrm{A}$, there are two depth images. One is projected from the Kinect V2 sensor, which is denoted as $\mathbf{D}'_k$. The other is the depth result of using the stereo matching method with the GoPro camera pair, which is expressed as $\mathbf{D}'_s$. The image resolutions of $\mathbf{D}'_k$ and $\mathbf{D}'_s$ are the same as that of the 4K GoPro camera $\mathbb{C}_\mathrm{A}$ or $\mathbb{C}_\mathrm{B}$. Let (i, j) be the coordinates of a 2D point on the camera image plane of $\mathbb{C}'_\mathrm{A}$, the fusion strategy for creating the final fused depth image $\mathbf{D}'_f$ is described as below:

$$\mathbf{D}'_f(i,\ j) = \begin{cases}\mathbf{D}'_k(i,\ j), & \text{if } \mathbf{D}'_k(i,\ j) > 0\ ;\\ \mathbf{D}'_s(i,\ j), & \text{else.}\end{cases} \tag{11}$$

The above depth fusion strategy is designed by considering the observation that the valid depth points in $\mathbf{D}'_k$ are normally denser than those in $\mathbf{D}'_s$. Besides, for this pixel-level depth fusion strategy, an accurate stereo-ToF calibration is the key to avoiding fusion artifacts in $\mathbf{D}'_f$.

4. Experiments

Experimental data are captured by the multi-camera rig device as illustrated in Fig. 1. An example image of the captured scene is presented in Fig. 5 (a). The details concerning the experimental parameter configuration and analysis are introduced as follows.

4.1. Experimental Settings

Image capture: A control computer is in charge of synchronizing the capture progress of the Kinect V2 sensor $\mathbb{C}_\mathrm{K}$ and two 4K GoPro cameras $\mathbb{C}_\mathrm{A}$, $\mathbb{C}_\mathrm{B}$ using the signal and time stamp technologies. The Kinect for Windows SDK is utilized to get the captured data from the Kinect V2 camera. Note that, this SDK is able to output color images and their corresponding registered depth images from the RGB sensor in $\mathbb{C}_\mathrm{K}$[1]. More specifically, both of these two

[1] Refer to the 'MapColorFrameToDepthSpace' method in the Kinect for Windows SDK.

(a) (b)

Figure 3. Detected corners of the $\mathbb{C}_{\mathrm{A}}$ view in (a) and of the $\mathbb{C}_{\mathrm{K}}$ view in (b).

Table I. The RMSE results of the stereo-ToF calibration using different coarse estimation approaches.

Coarse-to-Fine Framework	LHM	EPnP	RPnP	Ours
	Before depth correction			
Coarse estimation	1.82	3.14	1.45	0.87
Estimation Refinement	-	2.19	-	-
	After depth correction			
Coarse estimation	1.78	2.83	1.54	**0.87**
Estimation refinement	-	2.60	-	-

types of images used in experiments are in FHD resolution $(1,920 \times 1,080)$.

Camera Calibration: The intrinsic parameters and the radial and decentering distortion coefficients [2] of all the cameras on the multi-camera rig are measured by using a conventional checkerboard-based method [39]. The same calibration method is then exploited to estimate the extrinsic parameters of the GoPro camera pair in order to compute the rotation rectification matrices $(\mathbf{R}^a, \mathbf{R}^b)$, the intrinsic camera matrices $(\mathbf{K}'_a, \mathbf{K}'_b)$, and the perspective transformation matrix $\mathbf{Q}$ used for projecting the disparity map into the camera 3D space of $\mathbb{C}'_{\mathrm{A}}$. Note that there is no need to repeat this step every time because the two GoPro cameras have been fixed on the multi-camera rig.

Stereo-GoPro Depth: The Semi-Global Matching (SGM) algorithm is one of the most effective and efficient stereo matching methods [18], which is used here to estimate the disparity map of $\mathbb{C}'_{\mathrm{A}}$ with the rectified stereo images. The disparity map of $\mathbb{C}'_{\mathrm{A}}$ is projected into the camera 3D space of $\mathbb{C}'_{\mathrm{A}}$ with $\mathbf{Q}$ and then projected back onto the camera image plane of $\mathbb{C}'_{\mathrm{A}}$ to calculate $\mathbf{D}'_s$. Regarding the SGM method, the minimum disparity value is set to 256 and the maximum disparity value is set to $1,280$. The matched block size is equal to 9.

Point Pair Detection: The SIFT detector and descriptor algorithms are implemented by referring to their default implementations in OpenCV. The parameter r for ratio test is set to 0.8. The threshold ε of epipolar constraints in the RANSAC framework is set to 0.5 pixels.

Evaluation Metric: For the evaluation of the stereo-ToF calibration, a checkerboard appearing in both views of $\mathbb{C}_{\mathrm{A}}$ and $\mathbb{C}_{\mathrm{K}}$ is manually labeled at the locations of the common visible corners. Afterwards, a corner refinement approach with sub-pixel accuracy is applied to refine the positions of these corners [32]. As illustrated in Fig. 3, there are m $(= 47)$ common-corner point pairs in both views of $\mathbb{C}_{\mathrm{A}}$ and $\mathbb{C}_{\mathrm{K}}$, each of which is expressed as $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^k)$ as the description in section 3.3.1. When transforming the 2D point $\boldsymbol{u}_i^k$ to a 3D point $\boldsymbol{x}_i^k$ in the camera 3D space of $\mathbb{C}_{\mathrm{K}}$, the intensity-related distance error of the checkerboard in the Kinect V2 device is required to be considered [31, 20, 24]. To compensate the depth error of the corner point $\boldsymbol{u}_i^k$, a specific filter is adopted from [14]. Finally, the Root-Mean-Square Error (RMSE) metric is utilized to evaluate the error of the stereo-ToF calibration:

$$\sqrt{\frac{1}{m}\sum_{i=1}^{m} \|\boldsymbol{u}_i^a - \hat{\boldsymbol{u}}(\boldsymbol{x}_i^k, \mathbf{K}_a, \mathbf{R}_1, \boldsymbol{t}_1)\|^2}. \qquad (12)$$

For the evaluation of the depth fusion strategy, the Peak Signal-to-Noise Ratio (PSNR), Structural SIMilarity (SSIM) and Non-Black Region Proportion (NBRP) approaches are tried. In particular, the DIBR approach is exploited to render new views for $\mathbb{C}'_{\mathrm{B}}$ using the depth images $\mathbf{D}'_s, \mathbf{D}'_k, \mathbf{D}'_f$ respectively. The virtually-rendered images are then compared with the ground-truth images captured by $\mathbb{C}'_{\mathrm{B}}$ using PSNR, SSIM and NBRP metrics.

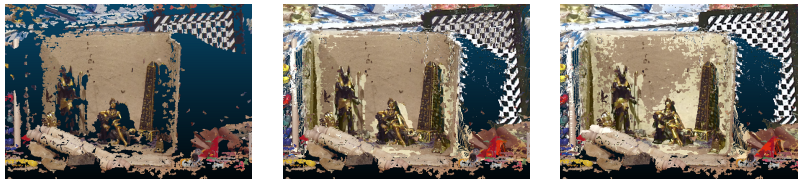
All experiments are conducted on an Intel Core i3-4030U laptop with 16 GB RAM and no GPU acceleration, using the captured datasets from the control computer. Both source code and datasets are going to be released on our website[2].

4.2. Results and Analysis

The proposed depth correction step, stereo-ToF calibration method and depth fusion strategy are analyzed quantitatively and qualitatively.

Quantitative Evaluation: The RMSE results of evaluating the precision of the stereo-ToF calibration methods are illustrated in Table I. Here, the symbol '-' indicates that the result of the estimation refinement step is worse than that of the coarse estimation step. In other words, the reliable point pairs detected by the approaches in section 3.2 make the calibration refinement algorithm get stuck in a local minimum. The calibration refinement algorithm works only for the case of the EPnP-based coarse estimation, while its performance is the worst among these four methods. The depth correction step slightly helps improving the RMSE results of LHM and EPnP for the coarse estimation stage. However, it has no influence on the calibration result of the proposed stereo-ToF calibration method using the camera orientation approximation. In summary, the proposed stereo-

[2] https://ygaokiel.github.io



(a) (b) (c)

Figure 4. The point clouds of a part of the scene in the camera 3D space of $\mathbb{C}'_{\mathrm{A}}$. The point cloud rendered with using the depth image $\mathbf{D}'_s$ is presented in (a). The point clouds from $\mathbf{D}'_s$ and $\mathbf{D}'_k$ without using the depth correction step are shown in (b). With using the depth correction step, the point clouds from $\mathbf{D}'_s$ and $\mathbf{D}'_k$ are exhibited in (c), in which the misalignment disappears compared with (b).

Table II. The performances of different depth sources for the rendering of virtual views using different evaluation metrics.

Depth Source	Before depth correction			After depth correction		
	PSNR	SSIM	NBRP (%)	PSNR	SSIM	NBRP (%)
$\mathbf{D}'_s$	9.29	0.156	61.29	-	-	-
$\mathbf{D}'_k$	11.29	0.235	74.50	11.35	0.245	74.28
$\mathbf{D}'_f$	11.98	0.247	**81.35**	**12.04**	**0.259**	81.15

ToF calibration method achieves the best performance compared with the other three baseline approaches.

The evaluation results of the depth fusion strategy are shown in Table II. The depth correction step improves the PSNR and SSIM values when using $\mathbf{D}'_k$ and $\mathbf{D}'_f$ for the virtual rendering, which indicates that the depth correction is a necessary step for depth accuracy of a Kinect V2 sensor. In addition, using $\mathbf{D}'_f$ from the depth fusion strategy achieves the best virtual-rendering performance in all of these three evaluation metrics, which exhibits the effectiveness of the proposed depth fusion strategy.

Qualitative Evaluation: To evaluate the effectiveness of the depth correction step, the point clouds with using the depth images $\mathbf{D}'_s, \mathbf{D}'_k$ are visualized in the camera 3D space of $\mathbb{C}'_{\mathrm{A}}$ as shown in Fig. 4. The rendering results without and with using the depth correction step are presented in Fig. 4 (b)(c). It can be found that, with using the depth correction step, the misalignment in the places of the golden tower and the checkerboard is eliminated, which indicates that the depth correction step contributes to the self-calibration of this stereo-ToF system.

As for the evaluation of the depth fusion strategy, the projected virtual images on the camera image plane of $\mathbb{C}'_{\mathrm{B}}$ using the depth images $\mathbf{D}'_s, \mathbf{D}'_k, \mathbf{D}'_f$ are shown in Fig. 5. The large black region near the bottom boarder of Fig. 5 (c) is mainly because the vertical FOV of the RGB sensor in the Kinect V2 camera is smaller than that of the GoPro camera $\mathbb{C}_{\mathrm{A}}$. The missing areas near the right boarders of Fig. 5 (b)(c) are caused by the camera displacement of $\mathbb{C}'_{\mathrm{A}}$ and $\mathbb{C}'_{\mathrm{B}}$. The image quality of the virtually projected image in Fig. 5 (b) is worse than that in Fig. 5 (c), which indicates that the Kinect V2 camera is more reliable than the GoPro camera pair for this specific scene. Moreover, the image quality of Fig. 5 (d) is better than those of Fig. 5 (b)(c), which shows that the proposed depth fusion strategy is an effective way of taking full advantage of both $\mathbf{D}'_s$ and $\mathbf{D}'_k$.

5. Conclusion

In this paper, a novel self-calibration method is proposed for a light-field movie capture device composed of a Kinect V2 and two 4K GoPro cameras. The proposed self-calibration method utilizes the geometric constraints in the scene and the cameras to overcome the disadvantage of the changeable tilt of the Kinect V2 camera. In addition, the camera orientation approximation step is used by our self-calibration method, which outperforms other baseline approaches. Moreover, a depth correction step is proven to be beneficial to the self-calibration of this stereo-ToF system. Furthermore, a depth fusion strategy is presented in this paper as well, which relies on the depth correction step and the rigid transformation result of the stereo-ToF calibration method, and is shown to be effective in rendering depth images of higher quality in 4K resolution.

6. Acknowledgments

The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, Intel-VCI-CAU, the German Research Foundation (DFG) No. K02044/8-1 and the Fraunhofer and Max Planck cooperation program within the German pact for research and innovation (PFI).

(a) Ground truth (Color image of $\mathbb{C}'_{\mathrm{B}}$)

(b) Projected virtual color image using $\mathbf{D}'_s$

(c) Projected virtual color image using $\mathbf{D}'_k$

(d) Projected virtual color image using $\mathbf{D}'_f$

Figure 5. Projected virtual color views on the camera image plane of $\mathbb{C}'_{\mathrm{B}}$ using different depth sources.

References

[1] N. S. Altman. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185, 1992. 3

[2] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971. 5

[3] A. Corti, S. Giancola, G. Mainetti, and R. Sala. A metrological characterization of the Kinect V2 time-of-flight camera. *Robotics and Autonomous Systems (RAS)*, 75:584–594, 2016. 1

[4] Ł. Dąbała, M. Ziegler, P. Didyk, F. Zilly, J. Keinert, K. Myszkowski, H.-P. Seidel, P. Rokita, and T. Ritschel. Efficient multi-image correspondences for on-line light field video processing. *Computer Graphics Forum*, 35(7):401–410, 2016. 1

[5] C. Dal Mutto, P. Zanuttigh, and G. M. Cortelazzo. A probabilistic approach to ToF and stereo data fusion. In *International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, pages 1–8, 2010. 2

[6] C. Dal Mutto, P. Zanuttigh, and G. M. Cortelazzo. Probabilistic ToF and stereo data fusion based on mixed pixels measurement models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(11):2260–2272, 2015. 2

[7] C. Dal Mutto, P. Zanuttigh, S. Mattoccia, and G. Cortelazzo. Locally consistent ToF and stereo data fusion. In *European Conference on Computer Vision Workshops (ECCVW)*, pages 598–607, 2012. 2

[8] G. D. Evangelidis, M. Hansard, and R. Horaud. Fusion of range and stereo data for high-resolution scene-modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 37(11):2178–2192, 2015. 2

[9] C. Fehn, R. De La Barré, and S. Pastoor. Interactive 3-DTV-concepts and key technologies. *Proceedings of the IEEE*, 94(3):524–538, 2006. 1

[10] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2, 3

[11] V. Gandhi, J. Čech, and R. Horaud. High-resolution depth maps based on ToF-stereo fusion. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4742–4749, 2012. 2

[12] Y. Gao, S. Esquivel, R. Koch, M. Ziegler, F. Zilly, and J. Keinert. A novel Kinect V2 registration method for large-displacement environments using camera and scene constraints. In *IEEE International Conference on Image Processing (ICIP)*, pages 997–1001, 2017. 2

[13] Y. Gao, M. Ziegler, F. Zilly, S. Esquivel, and R. Koch. A linear method for recovering the depth of Ultra HD cameras using a Kinect V2 sensor. In *IAPR International Conference on Machine Vision Applications (MVA)*, pages 494–497, 2017. 3

[14] V. Garro, C. Dal Mutto, P. Zanuttigh, and G. M. Cortelazzo. A novel interpolation scheme for range data with side information. In *Conference for Visual Media Production (CVMP)*, pages 52–60, 2009. 5

[15] S. Gauglitz, T. Höllerer, and M. Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *International Journal of Computer Vision (IJCV)*, 94(3):335–360, 2011. 3

[16] M. Hansard, G. Evangelidis, Q. Pelorson, and R. Horaud. Cross-calibration of time-of-flight and colour cameras. *Computer Vision and Image Understanding (CVIU)*, 134:105–115, 2015. 2

[17] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 3

[18] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(2):328–341, 2008. 5

[19] X. Hu and P. Mordohai. A quantitative evaluation of confidence measures for stereo vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(11):2121–2133, 2012. 1

[20] A. Kolb, E. Barth, R. Koch, and R. Larsen. Time-of-flight cameras in computer graphics. *Computer Graphics Forum*, 29(1):141–159, 2010. 5

[21] M. Kraft, M. Nowicki, A. Schmidt, M. Fularz, and P. Skrzypczyński. Toward evaluation of visual navigation algorithms on RGB-D data from the first-and second-generation Kinect. *Machine Vision and Applications (MVA)*, pages 1–14, 2016. 1

[22] V. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: An accurate O(n) solution to the PnP problem. *International Journal of Computer Vision (IJCV)*, 81(2):155–166, 2009. 2, 3

[23] S. Li, C. Xu, and M. Xie. A robust O(n) solution to the perspective-n-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(7):1444–1450, 2012. 2, 3

[24] M. Lindner, I. Schiller, A. Kolb, and R. Koch. Time-of-flight sensor calibration for accurate range sensing. *Computer Vision and Image Understanding (CVIU)*, 114(12):1318–1328, 2010. 4, 5

[25] M. I. A. Lourakis and A. A. Argyros. SBA: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software (TOMS)*, 36(1):2:1–2:30, 2009. 3

[26] K.-L. Low. Linear least-squares optimization for point-to-plane ICP surface registration. *Technical Report, Department of Computer Science, University of North Carolina at Chapel Hill*, TR04-004, 2004. 3

[27] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110, 2004. 3

[28] C.-P. Lu, G. D. Hager, and E. Mjolsness. Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(6):610–622, 2000. 2, 3

[29] G. Marin, P. Zanuttigh, and S. Mattoccia. Reliable fusion of ToF and stereo depth driven by confidence measures. In *European Conference on Computer Vision (ECCV)*, pages 386–401, 2016. 2

[30] R. Nair, K. Ruhl, F. Lenzen, S. Meister, H. Schäfer, C. S. Garbe, M. Eisemann, M. Magnor, and D. Kondermann. A survey on time-of-flight stereo fusion. In *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*, pages 105–127. 2013. 2

[31] H. Sarbolandi, D. Lefloch, and A. Kolb. Kinect range sensing: Structured-light versus time-of-flight Kinect. *Computer Vision and Image Understanding (CVIU)*, 139:1–20, 2015. 1, 4, 5

[32] D. Scaramuzza, A. Martinelli, and R. Siegwart. A toolbox for easily calibrating omnidirectional cameras. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5695–5701, 2006. 5

[33] M. Schmeing and X. Jiang. Depth image based rendering. In *Pattern Recognition, Machine Intelligence and Biometrics*, pages 279–310, 2011. 1

[34] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald. Review of stereo vision algorithms and their suitability for resource-limited systems. *Journal of Real-Time Image Processing*, 11(1):5–25, 2016. 1

[35] T.-C. Wang, A. A. Efros, and R. Ramamoorthi. Depth estimation with occlusion modeling using light-field cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(11):2170–2181, 2016. 1

[36] O. Wasenmüller and D. Stricker. Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision. In *Asian Conference on Computer Vision Workshops (ACCVW)*, pages 34–45, 2016. 1, 3, 4

[37] C. Zach. Robust bundle adjustment revisited. In *European Conference on Computer Vision (ECCV)*, pages 772–787, 2014. 4

[38] S. Zennaro, M. Munaro, S. Milani, P. Zanuttigh, A. Bernardi, S. Ghidoni, and E. Menegatti. Performance evaluation of the 1st and 2nd generation Kinect for multimedia applications. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2015. 1

[39] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22(11):1330–1334, 2000. 2, 5

[40] J. Zhu, L. Wang, J. Gao, and R. Yang. Spatial-temporal fusion for high accuracy depth maps using dynamic MRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 32(5):899–909, 2010. 2

[41] J. Zhu, L. Wang, R. Yang, and J. Davis. Fusion of time-of-flight depth and stereo for high accuracy depth maps. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 2

[42] S. Zinger, L. Do, and P. de With. Free-viewpoint depth image based rendering. *Journal of Visual Communication and Image Representation (JVCIR)*, 21(5):533–541, 2010. 1, 4

6.4 Publication 4

Parallax View Generation for Static Scenes Using Parallax-Interpolation Adaptive Separable Convolution

Yuan Gao and Reinhard Koch

Published in

2018 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), San Diego, CA, USA, 23-27 July 2018, pages 1-4.

DOI: 10.1109/ICMEW.2018.8551583

PARALLAX VIEW GENERATION FOR STATIC SCENES USING PARALLAX-INTERPOLATION ADAPTIVE SEPARABLE CONVOLUTION

Yuan Gao and Reinhard Koch

Christian-Albrechts-University of Kiel, 24118 Kiel, Germany

{yga, rk}@informatik.uni-kiel.de

ABSTRACT

Reconstructing a Densely-Sampled Light Field (DSLF) from a Sparsely-Sampled Light Field (SSLF) is a challenging problem, for which various kinds of algorithms have been proposed. However, very few of them treat the angular information in a light field as the temporal information of a video from a virtual camera, *i.e*. the parallax views of a SSLF for a static scene can be turned into the key frames of a video captured by a virtual camera moving along the parallax axis. To this end, in this paper, a novel parallax view generation method, Parallax-Interpolation Adaptive Separable Convolution (PIASC), is proposed. The presented PIASC method takes full advantage of the motion coherence of static objects captured by a SSLF device to enhance the motion-sensitive convolution kernels of a state-of-the-art video frame interpolation method, *i.e*. Adaptive Separable Convolution (AdaSepConv). Experimental results on three development datasets of the grand challenge demonstrate the superior performance of PIASC for DSLF reconstruction of static scenes.

***Index Terms*—** Parallax View Generation, View Synthesis, Densely-Sampled Light Field Reconstruction, Sparsely-Sampled Light Field Capture, Parallax-Interpolation Adaptive Separable Convolution

1. INTRODUCTION

Due to the rapidly growing market demand for the Virtual Reality (VR) [1] and Free Viewpoint Video (FVV) [2] contents, how to acquire a Densely-Sampled Light Field (DSLF) from a real-world static or dynamic scene for VR or FVV rendering is currently becoming a hot research topic. Moreover, DSLFs are capable of facilitating several computer vision applications that rely on light field data, *e.g*., depth estimation, super-resolution, synthetic aperture imaging, visual odometry, segmentation and compression [3]. However, building a DSLF capture system with a large number of densely-positioned cameras would be prohibitively expensive in data processing, camera synchronization and calibration. Therefore, sparse light field capture systems [4, 5, 6, 7, 8] have been designed for real-time light field video capture using a coarse set of cameras. How to reconstruct DSLFs from the sparse light fields captured by such systems is a challenging problem, which constitutes the main topic of the grand challenge on DSLF reconstruction.

To solve the DSLF reconstruction problem for a Sparsely-Sampled Light Field (SSLF) with parallax views for a static scene, a novel parallax view synthesis method, which is based on Adaptive Separable Convolution (AdaSepConv) [9], is proposed in this paper. Specifically, the proposed parallax view generation approach, Parallax-Interpolation Adaptive Separable Convolution (PIASC), applies a fine-tuning strategy to enhancing the convolution kernels of AdaSepConv with the consideration of the motion coherence of static objects in a parallax-view capture system. The PIASC method is evaluated on all the three development datasets of the grand challenge and further compared with AdaSepConv. Experimental results demonstrate the effectiveness of the proposed PIASC method and its superiority over the AdaSepConv approach for DSLF reconstruction of static scenes.

2. RELATED WORK

The DSLF reconstruction problem has been attempted to be solved by a lot of methods that can be generally categorized into several types, including light fields [10, 11], image-based rendering [12, 13], depth-image-based rendering [14, 15], optical flow [16, 17], light field angular super-resolution and learning-based view synthesis. Since the last two method categories are more related to the work in this paper, they are described in more detail below.

Light Field Angular Super-Resolution: Kalantari *et al*. propose a deep learning-based approach comprising both the color and disparity estimator components for view synthesis using a consumer light field camera [18]. More recently, light field angular super-resolution is explored in the area of angular detail restoration on Epipolar Plane Images (EPIs) of a light field with sparse parallax images. Vagharshakyan *et al*. present a DSLF reconstruction solution by dealing with the inpainting problem on EPIs using sparse regularization in shearlet transform domain, which is demonstrated to be effective in reconstructing non-Lambertian scenes containing semi-transparent objects [19, 20]. Wu *et al*. propose a blur-restoration-deblur framework for processing EPIs of a SSLF and restore the angular detail of the blurred and up-sampled EPIs with a Convolutional Neural Network (CNN) [21].

Learning-Based View Synthesis: Flynn *et al*. apply a deep architecture to addressing the view interpolation problem for

synthesizing novel natural imagery between wide baseline views in the real-world environments [22]. Niklaus *et al.* leverage a deep neural network to estimate spatially-adaptive 2D convolution kernels, which capture both the motion and interpolation information for pixel-wise video frame synthesis [23]. However, due to the large memory requirement for storing the convolution kernel information for all the image pixels, the whole desired virtual frame may not be synthesized at once by this method. To overcome this limitation, Niklaus *et al.* propose a spatially-adaptive separable convolution method by approximating each 2D convolution kernel with a pair of 1D kernels [9]. Liu *et al.* employ an end-to-end fully-convolutional deep network, Deep Voxel Flow (DVF), for video frame interpolation and extrapolation with sharp and realistic results [24].

3. METHODOLOGY

3.1. Preliminary

The parallax view generation of a SSLF for a static scene can be simplified as a novel view interpolation problem by utilizing the color information of only two RGB images from a pair of adjacent parallax views in this SSLF. Suppose the two RGB images are denoted by $\mathcal{I}_1$ and $\mathcal{I}_2$, and the intermediate view to be reconstructed is represented by $\tilde{\mathcal{I}}$. All these three images have the same resolution, $m \times n$ pixels. The AdaSepConv approach proposed in [9] is essentially to estimate two 2D convolution kernels, $\mathbf{K}_1(x, y)$ and $\mathbf{K}_2(x, y)$, for each 2D point (x, y) on $\tilde{\mathcal{I}}$. Specifically, each 2D kernel $\mathbf{K}_\mu(x, y)$ is approximated by a pair of 1D vectors $(\boldsymbol{v}^\mu_{x,y}, \boldsymbol{h}^\mu_{x,y})$:

$$\mathbf{K}_\mu(x, y) = \boldsymbol{v}^\mu_{x,y}(\boldsymbol{h}^\mu_{x,y})^{\mathrm{T}}. \tag{1}$$

The final color information for (x, y) on $\tilde{\mathcal{I}}$ is recovered by

$$\tilde{\mathcal{I}}(x, y, c) = \sum_{\mu=1}^{2} \big(\mathbf{K}_\mu(x, y) * \mathbf{P}_\mu(x, y, c)\big). \tag{2}$$

Here, '$*$' is the covolution operation symbol and c stands for the color channel, *i.e.* $c \in \{r, g, b\}$. Besides, $\mathbf{P}_\mu(x, y, c)$ represents the image patch centered at (x, y) in the c channel of $\mathcal{I}_\mu$, which has the same size as $\mathbf{K}_\mu(x, y)$, *i.e.* $k \times k$. Compared with the Adaptive Convolution (AdaConv) method proposed in [23], the AdaSepConv approach reduces the number of unknown kernel parameters from $2m \times n \times k^2$ to $2m \times n \times 2k$, thereby enabling a high-resolution synthesized view to be generated at once efficiently.

3.2. Parallax-Interpolation AdaSepConv (PIASC)

The DSLF reconstruction for a SSLF of a static scene can be treated as the frame interpolation of a standard video that is captured by a virtual camera moving along the parallax axis of a SSLF capture system. The AdaSepConv method is originally designed for novel frame synthesis for videos containing objects moving in different directions at varying speeds. Additionally, constructing a dedicated fully convolutional network based on AdaSepConv for the purpose of DSLF reconstruction is not always easy, considering that public high-resolution and high-quality real-world light field

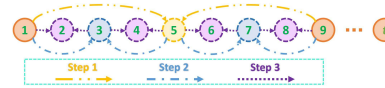


Fig. 1. A flow chart of reconstructing a DSLF from a sparse set of parallax views. The novel views are reconstructed recursively in three steps. The circles with solid lines represent duplicates of the under-sampled ground-truth camera views for the sub-challenge category $\mathcal{C}_1$ on a development dataset as described in Section 4.1. The circles with dash lines are unknown parallax views to be reconstructed.

datasets are not as common as public high-definition and high-fidelity real-world videos and the training process would be enormously time- and effort-consuming. In order to take full advantage of the state-of-the-art video frame interpolation method, AdaSepConv, for tackling the DSLF reconstruction problem and to avoid its cumbersome re-training process, a novel DSLF reconstruction method, PIASC, is proposed. More details about it are introduced as below.

A 2D convolution kernel $\mathbf{K}_\mu(x, y)$ generated by the deep neural network of AdaSepConv contains both motion and resampling information for any object moving in any direction. However, in the grand challenge on DSLF reconstruction, a SSLF dataset is composed of a sparse set of parallax images for a static scene; in other words, static objects in these parallax images have only one motion direction that coincides with the parallax axis of the SSLF dataset. Intuitively, performing a fine-tuning strategy that enhances the motion-sensible convolution kernels of AdaSepConv should be beneficial to the parallax view synthesis for a SSLF. Accordingly, the proposed PIASC method implements this fine-tuning process by adjusting all the coefficient values in the convolution kernels with an elaborately designed weight matrix $\mathbf{W}$, *i.e.*

$$\hat{\mathbf{K}}_\mu(x, y) = \frac{k^2}{\sum_{i=1}^{k}\sum_{j=1}^{k} \mathbf{W}(i, j; \sigma)} \mathbf{W} \circ \mathbf{K}_\mu(x, y), \tag{3}$$

where

$$\mathbf{W}(i, j; \sigma) = \exp\left(-\frac{1}{2}\left(\frac{|j - \bar{k}|}{\sigma}\right)^2\right), \quad i, j \in \mathbb{Z} \cap [1, k], \;\; \bar{k} = \frac{k+1}{2}. \tag{4}$$

Here, '$\circ$' denotes the element-wise (Hadamard) product and $\hat{\mathbf{K}}_\mu(x, y)$ represents the horizontal-motion-enhanced convolution kernel generated by PIASC for the DSLF reconstruction along the horizontal parallax axis of a SSLF. The weight matrix $\mathbf{W}$ is similar to a Gaussian kernel; however, only the coordinate information along the vertical axis of $\mathbf{W}$ is taken into account by PIASC, which is because of the horizontal-parallax feature of the datasets in the grand challenge.

3.3. DSLF Reconstruction for SSLFs

After the above introduction about the proposed PIASC method, this section is dedicated to investigating how to lever-

Input: Dataset size t $(= 193)$;
Camera view interval $\tau \in \{8, 16, 32\}$;
Camera view $\mathcal{I}_\mu$, $\mu \in \{1, 1+\tau, 1+2\tau, \ldots, t\}$.
Output: Reconstructed view $\tilde{\mathcal{I}}_\omega$, $\omega \in \mathbb{Z}^+$ and $\omega \leqslant t$.

/* range $(1,\ t,\ \tau) = \{1, 1+\tau, 1+2\tau, \ldots, t\}$ */
for ω **in** range $(1,\ t,\ \tau)$ **do**
  $\tilde{\mathcal{I}}_\omega \leftarrow \mathcal{I}_\omega$;
end
while $\tau > 1$ **do**
  $\dot{\tau} \leftarrow \frac{\tau}{2}$;
  for ω **in** range $(1,\ t-\tau,\ \tau)$ **do**
    $\tilde{\mathcal{I}}_{\omega+\dot{\tau}} \leftarrow \text{PIASC}(\tilde{\mathcal{I}}_\omega, \tilde{\mathcal{I}}_{\omega+\tau})$;
  end
  $\tau \leftarrow \dot{\tau}$;
end

Algorithm 1: A parallax view generation algorithm for a SSLF dataset, which is created from a DSLF dataset.

age this approach to reconstruct a DSLF from a SSLF. The overall process of DSLF reconstruction for a SSLF is depicted in Algorithm 1. The camera view interval τ denotes the sampling interval on a DSLF dataset comprising ground-truth parallax views. The under-sampled parallax views in this DSLF dataset form a SSLF dataset, which is firstly used to recover a portion of parallax views of a desired unknown DSLF. The orange circles with solid lines illustrated in Fig. 1 stand for these reconstructed views, which are essentially duplicates of all the views in the SSLF dataset. The unknown views in the middle of adjacent reconstructed views are then synthesized by utilizing the PIASC method, corresponding to the Step 1 and yellow dash-line circle '5' in Fig. 1. Finally, this operation is repeated recursively until all the parallax views of the desired unknown DSLF are reconstructed, *i.e.* the Step 2 and 3 in Fig. 1.

4. EXPERIMENTS

4.1. Experimental Settings

Grand Challenge Introduction: The grand challenge on DSLF reconstruction has three sub-challenges, *i.e.* three categories of decimated-parallax imagery for DSLF reconstruction, which are denoted by $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$. In particular, these three categories have different numbers of camera views along parallax axis, such that the adjacent images in $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$ have varying disparity ranges, *i.e.* narrow (≈ 8 pixels), moderate ($\approx 15-16$ pixels), and wide ($\approx 30-32$ pixels), corresponding to $\tau = 8, 16, 32$ in Algorithm 1 separately.

Development Datasets: To evaluate the performance of different DSLF reconstruction algorithms, three Development Datasets (DDs) of varing 3D scenes are provided by the grand challenge. The development datasets are composed of pre-rectified horizontal-parallax multi-perspective RGB images with the same resolution, *i.e.* $m = 1,280$ and $n = 720$ in Section 3.1. Two of the datasets, Lambertian DD and Complex DD, are captured by a high-quality low-noise camera mounted on a highly-precise gantry for two different real 3D scenes. The third one, Synthetic DD, is rendered by Blender for a photorealistic 3D scene.

Table I. The lowest per-view PSNR results (in dB, explained in Section 4.1) for the performance evaluation of different methods on three development datasets for three sub-challenge categories of the grand challenge.

Lambertian DD				
Cat.	AdaSepConv ($\mathcal{L}_1$)	AdaSepConv ($\mathcal{L}_F$)	PIASC ($\mathcal{L}_1$)	PIASC ($\mathcal{L}_F$)
$\mathcal{C}_1$	43.001	43.162	**44.253**	43.657
$\mathcal{C}_2$	41.619	41.708	**43.091**	42.160
$\mathcal{C}_3$	38.857	38.760	**38.988**	38.436
Synthetic DD				
Cat.	AdaSepConv ($\mathcal{L}_1$)	AdaSepConv ($\mathcal{L}_F$)	PIASC ($\mathcal{L}_1$)	PIASC ($\mathcal{L}_F$)
$\mathcal{C}_1$	36.329	36.156	**36.451**	36.186
$\mathcal{C}_2$	35.256	35.143	**35.491**	35.271
$\mathcal{C}_3$	32.539	32.312	**32.666**	32.333
Complex DD				
Cat.	AdaSepConv ($\mathcal{L}_1$)	AdaSepConv ($\mathcal{L}_F$)	PIASC ($\mathcal{L}_1$)	PIASC ($\mathcal{L}_F$)
$\mathcal{C}_1$	34.620	34.682	**34.736**	34.645
$\mathcal{C}_2$	30.884	30.897	**30.974**	30.866
$\mathcal{C}_3$	27.500	26.922	**27.538**	26.896
Average performance for each sub-challenge category across all the DDs				
Cat.	AdaSepConv ($\mathcal{L}_1$)	AdaSepConv ($\mathcal{L}_F$)	PIASC ($\mathcal{L}_1$)	PIASC ($\mathcal{L}_F$)
$\mathcal{C}_1$	37.983	38.000	**38.480**	38.163
$\mathcal{C}_2$	35.920	35.916	**36.519**	36.099
$\mathcal{C}_3$	32.965	32.665	**33.064**	32.555

Evaluation Criteria: The reconstructed parallax views for each sub-challenge category on a develop dataset are compared against ground-truth horizontal-parallax images in this develop dataset. The per-view PSNR is exploited to perform the quality evaluation for a reconstructed view $\tilde{\mathcal{I}}$ with using its corresponding ground-truth view $\mathcal{I}$, *i.e.*

$$\text{MSE} = \frac{1}{m \times n \times 3} \sum_{x=1}^{m} \sum_{y=1}^{n} \left\| \tilde{\mathcal{I}}(x,y) - \mathcal{I}(x,y) \right\|_2^2, \qquad \text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right). \tag{5}$$

The lowest per-view PSNR for a sub-challenge category on a development dataset is selected as the single quality measure for this category on this development dataset.

Implementation Details: The pre-trained fully convolutional neural networks of AdaSepConv with $\mathcal{L}_1$ loss and perceptual loss $\mathcal{L}_F$ are from [9]. Regarding the parameters for the weight matrix $\mathbf{W}$ in (4), $k = 51$ and $\sigma = 200$.

4.2. Results and Analysis

The quantitative evaluation results of the proposed methods and baseline approaches are exhibited in Table I. As can be seen from this table, for the same sub-challenge category, all the methods achieve the best performance on the Lambertian DD, but the worst performance on the Complex DD. This is because the real 3D scene of Lambertian DD consists of objects with Lambertian reflectance only; however, the photo-realistic 3D scene of Synthetic DD has predominantly semi-transparent objects and the real 3D scene of Complex DD includes depth variations, occlusions and reflective objects. In other words, the scene-complexity order for the development datasets is Lambertian DD $<$ Synthetic DD $<$ Complex DD.

Besides, the proposed PIASC method with $\mathcal{L}_1$ loss outperforms the other three approaches on all the three development datasets for each sub-challenge category, which proves the effectiveness of PIASC ($\mathcal{L}_1$) method for DSLF reconstruction of static scenes. Moreover, the average performance for all the methods on each sub-challenge category across all the three development datasets is shown at the bottom of Table I. It can be found that, for $\mathcal{C}_1$, $\mathcal{C}_2$, $\mathcal{C}_3$, the PIASC ($\mathcal{L}_1$) method achieves average-PSNR improvements of 1.26%, 1.67% and 0.30% compared to the maximal average-PSNR values of AdaSepConv ($\mathcal{L}_1$) and AdaSepConv ($\mathcal{L}_F$) on these three sub-challenge categories. It implies that the proposed PIASC ($\mathcal{L}_1$) approach is more effective for DSLF reconstruction of static scenes with moderate occlusions and specular reflections.

5. CONCLUSION

This paper presents a novel parallax view generation algorithm based on PIASC for the grand challenge on DSLF reconstruction for decimated-parallax imagery of static scenes. The proposed PIASC method fully leverages the object-motion coherence of a horizontal-parallax SSLF to enhance the motion-sensitive convolution kernels, which are generated by one of the state-of-the-art learning-based video frame synthesis approaches, *i.e.* AdaSepConv. Experimental results on three development datasets with varying level of scene complexity show that PIASC achieves the better DSLF reconstruction performance than the AdaSepConv approach.

6. ACKNOWLEDGMENTS

The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, the German Research Foundation (DFG) No. K02044/8-1. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

7. REFERENCES

[1] J. Yu, "A light-field journey to virtual reality," *IEEE MultiMedia*, vol. 24, no. 2, pp. 104–112, 2017. 1

[2] A. Smolic, "3D video and free viewpoint video - From capture to display," *Pattern Recognition*, vol. 44, no. 9, pp. 1958–1968, 2011. 1

[3] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field image processing: An overview," *IEEE J-STSP*, vol. 11, no. 7, pp. 926–954, 2017. 1

[4] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," *ACM TOG*, vol. 24, no. 3, pp. 765–776, 2005. 1

[5] L. Dąbała, M. Ziegler, P. Didyk, F. Zilly, J. Keinert, K. Myszkowski, H.-P. Seidel, P. Rokita, and T. Ritschel, "Efficient multi-image correspondences for on-line light field video processing," *Computer Graphics Forum*, vol. 35, no. 7, pp. 401–410, 2016. 1

[6] S. Esquivel, Y. Gao, T. Michels, L. Palmieri, and R. Koch, "Synchronized data capture and calibration of a large-field-of-view moving multi-camera light field rig," in *3DTV-CON Workshops*, 2016. 1

[7] Y. Gao, S. Esquivel, R. Koch, M. Ziegler, F. Zilly, and J. Keinert, "A novel Kinect V2 registration method for large-displacement environments using camera and scene constraints," in *ICIP*, 2017, pp. 997–1001. 1

[8] Y. Gao, S. Esquivel, R. Koch, and J. Keinert, "A novel self-calibration method for a stereo-ToF system using a Kinect V2 and two 4K GoPro cameras," in *3DV*, 2017. 1

[9] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *ICCV*, 2017, pp. 261–270. 1, 2, 3

[10] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*, 1996, pp. 31–42. 1

[11] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *SIGGRAPH*, 1996, pp. 43–54. 1

[12] H.-Y. Shum, S.-C. Chan, and S.-B. Kang, *Image-based rendering*, Springer Science+Business Media, 2007. 1

[13] H.-Y. Shum, S.-B. Kang, and S.-C. Chan, "Survey of image-based representations and compression techniques," *IEEE TCSVT*, vol. 13, no. 11, pp. 1020–1037, 2003. 1

[14] C. Zhu, Y. Zhao, L. Yu, and M. Tanimoto, *3D-TV system with depth-image-based rendering*, Springer Science+Business Media, 2013. 1

[15] C. Fehn, R. De La Barré, and S. Pastoor, "Interactive 3-DTV-concepts and key technologies," *Proceedings of the IEEE*, vol. 94, no. 3, pp. 524–538, 2006. 1

[16] L. Xu, J. Jia, and Y. Matsushita, "Motion detail preserving optical flow estimation," *IEEE TPAMI*, vol. 34, no. 9, pp. 1744–1757, 2012. 1

[17] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *IJCV*, vol. 92, no. 1, pp. 1–31, 2011. 1

[18] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM TOG*, vol. 35, no. 6, pp. 193:1–193:10, 2016. 1

[19] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018. 1

[20] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017. 1

[21] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field reconstruction using deep convolutional network on EPI," in *CVPR*, 2017, pp. 1638–1646. 1

[22] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "DeepStereo: Learning to predict new views from the world's imagery," in *CVPR*, 2016, pp. 5515–5524. 2

[23] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *CVPR*, 2017, pp. 2270–2279. 2

[24] Z. Liu, R. Yeh, X. Tang, Y. Liu, and A. Agarwala, "Video frame synthesis using deep voxel flow," in *ICCV*, 2017, pp. 4473–4481. 2

6.5 Publication 5

A Novel Kinect V2 Registration Method Using Color and Deep Geometry Descriptors

Yuan Gao, Tim Michels and Reinhard Koch

Published in

2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3-7 Sept. 2018, pages 201-205.

DOI: 10.23919/EUSIPCO.2018.8553327

A Novel Kinect V2 Registration Method Using Color and Deep Geometry Descriptors

Yuan Gao, Tim Michels and Reinhard Koch
Christian-Albrechts-University of Kiel, 24118 Kiel, Germany
{yga, tmi, rk}@informatik.uni-kiel.de

***Abstract*—The novel view synthesis for traditional sparse light field camera arrays generally relies on an accurate depth approximation for a scene. To this end, it is preferable for such camera-array systems to integrate multiple depth cameras (*e.g.* Kinect V2), thereby requiring a precise registration for the integrated depth sensors. Methods based on special calibration objects have been proposed to solve the multi-Kinect V2 registration problem by using the prebuilt geometric relationships of several easily-detectable common point pairs. However, for registration tasks incapable of knowing these precise geometric relationships, this kind of method is prone to fail. To overcome this limitation, a novel Kinect V2 registration approach in a coarse-to-fine framework is proposed in this paper. Specifically, both local color and geometry information is extracted directly from a static scene to recover a rigid transformation from one Kinect V2 to the other. Besides, a 3D convolutional neural network (ConvNet), *i.e.* 3DMatch, is utilized to describe local geometries. Experimental results show that the proposed Kinect V2 registration method using both color and deep geometry descriptors outperforms the other coarse-to-fine baseline approaches.**

I. Introduction

The second version of the Microsoft Kinect (Kinect V2) is one of the most widespread low-cost Time-of-Flight (ToF) sensors available in the market [1]. The comparison between the Kinect V2 and the first generation of Microsoft Kinect (Kinect V1) is well studied in [2], where the Kinect V2 has a higher accuracy but a lower precision than the Kinect V1 [3].

A. Motivation

The multi-camera rig illustrated in Fig. 1 (a) is a movable camera array [4] for capturing dynamic light fields [5]. The precise calibration of the two Kinect V2 sensors on this rig is critical to the dense 3D reconstruction of a large-scale and non-rigid scene [6], which can be further used for the novel view synthesis in the Free Viewpoint Video (FVV) [7] and Head-Mounted Display (HMD) [8] systems, together with the dynamic light fields captured by the sparse RGB camera array and densely reconstructed by [9]–[14]. Therefore, an automatic Kinect V2 registration method without relying on any calibration object would be highly desirable for this system, considering that the positions of the two Kinect V2 cameras may be changed for different scenes of varying sizes and the preparation phase of calibration object-based registration methods may be time-consuming and cumbersome.

B. Related Work

As for solving the registration problem of multiple depth cameras with using calibration objects, several methods have been proposed. Afzal *et al.* propose an RGB-D multi-view system calibration method, *i.e.* BAICP+, which combines

(a) A multi-camera system. (b) A static scene.

Figure 1. The two Kinect V2 cameras are fixed on a movable multi-camera rig. The static scene shown in (b) is used for experiments.

Bundle Adjustment (BA) [15] and Iterative Closest Point (ICP) [16] into a single minimization framework [17]. The corners of a checkerboard are detected for the BA part of BAICP+. Kowalski *et al.* present a coarse-to-fine solution for the multi-Kinect V2 calibration problem, where a planar marker is used for the rough estimation of camera poses, which is later refined by an ICP algorithm [18]. Soleimani *et al.* employ three double-sided checkerboards placed at varying depths for an automatic calibration process of two opposing Kinect V2 cameras [19]. Córdova-Esparza *et al.* introduce a calibration tool for multiple Kinect V2 sensors using a 1D calibration object, *i.e.* a wand, which has three collinear points [20]. Regarding the Kinect V2 registration solution without using calibration objects, Gao *et al.* propose a coarse-to-fine Kinect V2 calibration approach using camera and scene constraints for two Kinect V2 cameras with a large displacement [21].

In this paper, to solve the registration problem of two Kinect V2 cameras, a novel camera calibration method for Kinect V2 sensors using local color and geometry information is proposed. Specifically, an off-the-shelf feature detector is used for detecting interest points and describing local color information for them. Afterwards, a ConvNet-based 3D descriptor, 3DMatch [22], is utilized to describe local geometry information for these interest points. Both color and geometry descriptors are employed to estimate an initial rough rigid transformation between two Kinect V2 cameras, which can then be refined by an optional estimation refinement step if necessary. Experimental results prove the effectiveness of the proposed method by comparing it with baseline approaches.

II. Methodology

A. Preliminary

The two Kinect V2 cameras mounted on the multi-camera rig are denoted by $\mathbb{C}_\mathrm{A}$ and $\mathbb{C}_\mathrm{B}$, respectively. Since the intrinsic parameters and lens distortion of the ToF sensor in a Kinect V2 can be calibrated in advance or extracted from the factory calibration by using the Kinect for Windows SDK, the

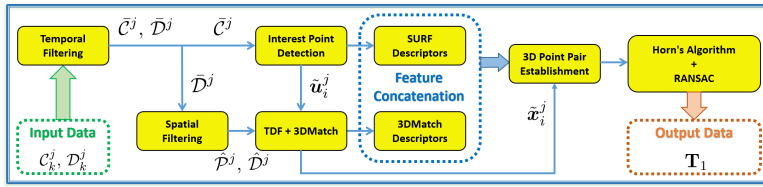


Figure 2. A flow chart of the proposed Kinect V2 registration method in the coarse estimation phase.

registration problem of two Kinect V2 cameras is interpreted as how to calculate a rigid transformation between these two cameras. Suppose a rigid transformation is expressed as

$$\mathbf{T}_i = \begin{pmatrix} \mathbf{R}_i & \boldsymbol{t}_i \\ \mathbf{0} & 1 \end{pmatrix} \in \mathbb{SE}(3), \tag{1}$$

where $\mathbf{R}_i \in \mathbb{SO}(3)$ and $\boldsymbol{t}_i \in \mathbb{R}^3$. A coarse-to-fine camera registration framework [23], [24] is defined as estimating the rigid transformation from $\mathbb{C}_\mathrm{A}$ to $\mathbb{C}_\mathrm{B}$ via two steps:

$$\mathbf{T} = \mathbf{T}_2\mathbf{T}_1. \tag{2}$$

Here, for the rigid transformation matrix $\mathbf{T}_i$, $i \in \{1, 2\}$ stands for the case of using a coarse estimation method in Section II-B and the other case of using an estimation refinement approach in Section II-C, respectively. The camera coordinate system of the ToF sensor in a Kinect V2 camera is specified as the camera space of this Kinect V2. The intrinsic parameters of $\mathbb{C}_\mathrm{A}$ or $\mathbb{C}_\mathrm{B}$ are represented by the focal lengths f_x^j, f_y^j and the principal point $\left(c_x^j, c_y^j\right)^\mathrm{T}$, where $j \in \{a, b\}$. The lens distortion coefficients are utilized to eliminate distortions before saving any pair of registered color and depth images, denoted by $\mathcal{C}^j$ and $\mathcal{D}^j$, both of which are from the camera image plane of the ToF sensor in a Kinect V2.

B. Coarse Estimation

A marker that can be simultaneously captured by a pair of Kinect V2 cameras aids establishing reliable corresponding 2D point pairs on $\mathcal{C}^a$ and $\mathcal{C}^b$ [18]. One of these corresponding 2D point pairs is expressed as $\left(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b\right)$, where $\boldsymbol{u}_i^j = \left(u_i^j, v_i^j, 1\right)^\mathrm{T}$, $j \in \{a, b\}$. The depth value d_i^j for a 2D point $\boldsymbol{u}_i^j$ is acquired from its respective depth image $\mathcal{D}^j$, *i.e.* $d_i^j = \mathcal{D}^j\left(v_i^j, u_i^j\right)$. Defining $s : \mathbb{R}^3 \times \mathbb{R} \to \mathbb{R}^4$ to be a back-projection function, which projects a 2D point $\boldsymbol{u}_i^j$ on the camera image plane to a 3D point $\boldsymbol{x}_i^j = \left(x_i^j, y_i^j, z_i^j, 1\right)^\mathrm{T}$ in the camera space,

$$s(\boldsymbol{u}_i^j, d_i^j) = \left(\frac{(u_i^j - c_x^j)d_i^j}{f_x^j}, \quad \frac{(v_i^j - c_y^j)d_i^j}{f_y^j}, \quad d_i^j, \quad 1 \right)^\mathrm{T}, \tag{3}$$

and $\boldsymbol{x}_i^j = s(\boldsymbol{u}_i^j, d_i^j)$. The 2D point pair $\left(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b\right)$ is therefore able to be turned into a 3D point pair $\left(\boldsymbol{x}_i^a, \boldsymbol{x}_i^b\right)$ by (3). The coarse rigid transformation $\mathbf{T}_1$ is estimated by

$$\arg \min_{\mathbf{R}_1 \in \mathbb{SO}(3), \boldsymbol{t}_1 \in \mathbb{R}^3} \sum_{i=1}^{n} \frac{1}{2} \left\| \begin{pmatrix} \mathbf{R}_1 & \boldsymbol{t}_1 \\ \mathbf{0} & 1 \end{pmatrix} \boldsymbol{x}_i^a - \boldsymbol{x}_i^b \right\|_2^2. \tag{4}$$

The minimization problem in (4) can be turned into the Orthogonal Procrustes problem [25] and solved by the least-squares fitting algorithm [26] efficiently, requiring at least three corresponding 3D point pairs, *i.e.* $n \geqslant 3$.

However, preparing some special calibration objects for the Kinect V2 registration task is sometimes time- and effort-consuming. How to solve the Kinect V2 registration problem by only using the information from a nature scene is more challenging than the above case of using a marker. To deal with this problem, a novel coarse estimation framework is proposed and presented in Fig. 2. This framework exploits both color and geometry feature descriptors to estimate a rough rigid transformation $\mathbf{T}_1$ between two Kinect V2 cameras. Details about it are described as below:

1) Input Data: Due to the precision problem [3] of the ToF sensor of any Kinect V2, multi-frame depth information is used to improve the quality of the captured depth images. For a static scene and a static multi-camera system, m consecutive depth and color frames are captured by both $\mathbb{C}_\mathrm{A}$ and $\mathbb{C}_\mathrm{B}$ simultaneously. The input data for the coarse estimation framework are $\mathcal{C}_k^j$ and $\mathcal{D}_k^j$, where $k \in \mathbb{Z}^+$, $k \leqslant m$, and $j \in \{a, b\}$.

2) Temporal Filtering: A temporal mean filter is used here to calculate an average depth image $\bar{\mathcal{D}}^j$ for all the $\mathcal{D}_k^j$ images. Note that an underlying depth-validity check is also performed by this depth temporal mean filter. In particular, only depth image pixels with depth values larger than $0.5\,\mathrm{m}$ are treated as valid pixels for the accumulated weights. A corresponding average color image $\bar{\mathcal{C}}^j$ is accordingly generated by using all the $\mathcal{C}_k^j$ images and the same accumulated weights with valid pixel positions from the depth temporal filtering process.

3) Spatial Filtering: The mean depth image $\bar{\mathcal{D}}^j$ is then projected into a point cloud $\bar{\mathcal{P}}^j$ in the camera space of the Kinect V2 by using (3). However, the resulting point cloud $\bar{\mathcal{P}}^j$ may still have some outliers or noisy data, some of which are far away from the real captured scene. This will increase the volume allocation for the volumetric representation in the following steps, which may lead to a failure if limited memory is available in hardware, *e.g.* GPU. To handle this problem, a statistical spatial filtering method is utilized to trim the outliers of $\bar{\mathcal{P}}^j$. To be precise, each 3D point $\boldsymbol{x}_i^j$ in this point cloud has a mean distance t_i^j to its l nearest neighbor 3D points. A 3D point $\boldsymbol{x}_i^j$ will be removed if its distance t_i^j is not inside the range determined by the global distances mean and standard deviation. The filtered point cloud is denoted by $\hat{\mathcal{P}}^j$ and projected back onto the camera image plane by using

$$\pi(\boldsymbol{x}_i^j) = \left(\frac{f_x^j x_i^j}{z_i^j} + c_x^j, \quad \frac{f_y^j y_i^j}{z_i^j} + c_y^j, \quad 1 \right)^\mathrm{T}, \tag{5}$$

which generates a filtered depth image $\hat{\mathcal{D}}^j$ accordingly.

Algorithm 1: An ICP-based estimation refinement algorithm.

Input : $\hat{\mathcal{P}}^j$ from Section II-B3, Rigid transformation $\mathbf{T}_1$.
Output: Rigid transformation $\mathbf{T}_2$.

```
/* Step 1: Transform P̂^a from C_A to C_B coordinates    */
foreach point x_i^a in P̂^a do x_i^a ← T_1 x_i^a;
/* Step 2: Point cloud registration                      */
τ ← 0.005;
e ← +∞, ē ← 0, ė ← 0;
T^a ← I_4, T^b ← I_4, Ṫ ← I_4; /* I_n: n × n identity matrix */
while true do
    Ṫ^a ← T^a;
    Ṫ^b ← T^b;
    ė ← e;
    e ← 0;
    Ṫ, ė ← ICP(P̂^a, P̂^b); /* ė: Average error per point */
    foreach point x_i^a in P̂^a do x_i^a ← Ṫ x_i^a;
    T^a ← Ṫ T^a;
    e ← e + ė;
    Ṫ, ė ← ICP(P̂^b, P̂^a);
    foreach point x_i^b in P̂^b do x_i^b ← Ṫ x_i^b;
    T^b ← Ṫ T^b;
    e ← e + ė;
    if e > ē then
        T^a ← Ṫ^a;
        T^b ← Ṫ^b;
        break;
    if (ė − e)/e < τ then break;
T_2 ← (T^b)^{-1} T^a.
```

4) Interest Point Detection: The Speeded Up Robust Features (SURF) have robust and stable performance in computer vision and robotics applications [27]. The SURF interest point detector is used to detect 2D keypoints on the average color image $\bar{\mathcal{C}}^j$ from the temporal filtering step (Section II-B2). The coordinates of all the keypoints are fed to the next step for geometry feature calculation. Besides, for each detected 2D interest point $\tilde{\boldsymbol{u}}_i^j$, the SURF algorithm also generates a SURF descriptor $\tilde{\boldsymbol{\omega}}_i^j \in \mathbb{R}^{64}$, which is a normalized vector.

5) TDF and 3DMatch: The Truncated Distance Function (TDF) representation is a variation of Truncated Signed Distance Function (TSDF) [28]. The filtered point cloud $\hat{\mathcal{P}}^j$ is assigned to a volumetric grid of voxels to calculate the TDF value for each voxel. As for each 2D interest point $\tilde{\boldsymbol{u}}_i^j$, a corresponding 3D interest point $\tilde{\boldsymbol{x}}_i^j$ is computed by (3) with its depth information from $\hat{\mathcal{D}}^j$. A volumetric 3D patch for each $\tilde{\boldsymbol{x}}_i^j$ is then extracted from the volumetric grid, *i.e.*, $\tilde{\boldsymbol{x}}_i^j$ is in the center of a $30 \times 30 \times 30$ local voxel grid. The extracted volumetric 3D patch is finally fed into a pre-trained network of 3DMatch to generate a local geometry descriptor $\tilde{\boldsymbol{\epsilon}}_i^j \in \mathbb{R}^{512}$.

6) Feature Concatenation: To make full use of different advantages of the SURF and 3DMatch descriptors for the scene representation, a feature concatenation strategy is proposed as below:

$$\tilde{\boldsymbol{\rho}}_i^j = (1-\lambda)\tilde{\boldsymbol{\omega}}_i^j \oplus \lambda\tilde{\boldsymbol{\epsilon}}_i^j = \begin{pmatrix} (1-\lambda)\tilde{\boldsymbol{\omega}}_i^j \\ \lambda\tilde{\boldsymbol{\epsilon}}_i^j \end{pmatrix}, \lambda \in [0,1]. \quad (6)$$

The resulting concatenated descriptor is denoted by $\tilde{\boldsymbol{\rho}}_i^j \in \mathbb{R}^{576}$.

7) 3D Point Pair Establishment: After constructing the concatenated feature descriptor $\tilde{\boldsymbol{\rho}}_i^j$ for each 3D interest point $\tilde{\boldsymbol{x}}_i^j$, the reliable corresponding 3D point pairs in the two Kinect V2 camera spaces are established by means of the k-d tree data structure [29] and k-Nearest-Neighbors algorithm [30].

(a) Average color image $\bar{\mathcal{C}}^a$. (b) Average color image $\bar{\mathcal{C}}^b$.
Figure 3. The average color images from the temporal filtering step (Section II-B2). Green circles and red crosses stand for the corners of check patterns.

8) Horn's Algorithm and RANSAC: The final rigid transformation $\mathbf{T}_1$ from $\mathbb{C}_\mathrm{A}$ to $\mathbb{C}_\mathrm{B}$ for the coarse estimation step is calculated by using the Horn's algorithm [31] together with the RANdom SAmple Consensus (RANSAC) method [32] for solving the least squares problem defined in (4).

C. Estimation Refinement

The algorithm for estimation refinement is depicted in Algorithm 1. The input data for this algorithm are the rough rigid transformation $\mathbf{T}_1$ of the previous coarse estimation stage and point clouds $\hat{\mathcal{P}}^a$ and $\hat{\mathcal{P}}^b$ from the spatial filtering step (Section II-B3). The point cloud $\hat{\mathcal{P}}^a$ is firstly transformed into the camera coordinate system of $\mathbb{C}_\mathrm{B}$. Afterwards, the two point clouds in the same camera space are registered by using an ICP-based method, which in this case is equal to the camera pose refinement. The final estimation refinement result $\mathbf{T}_2$ is recovered from two intermediate rigid transformation matrices $\mathbf{T}^a$ and $\mathbf{T}^b$.

III. Experiments

A. Experimental Settings

1) Camera Setup: The equipment for capturing experimental data is a multi-camera system as shown in Fig. 1 (a). This system has two Kinect V2 cameras with similar orientations. The horizontal displacement between them is around 1.5 m. The Kinect for Windows SDK is leveraged to capture a static scene for both $\mathbb{C}_\mathrm{A}$ and $\mathbb{C}_\mathrm{B}$. The intrinsic parameters f_x^j, f_y^j, c_x^j, c_y^j and radial distortion coefficients [33] are extracted from the hardware of Kinect V2 sensors by using this SDK.

2) Static Scene: An example image of the static scene is exhibited in Fig. 1 (b). The positions of check patterns in the scene are adopted in the following evaluation metric step. The size of this scene is $5.5 \times 3.0 \times 3.6\,\mathrm{m}^3$ ($w \times h \times d$). The number of captured color or depth frames, *i.e.* m in Section II-B1, is equal to 31. The average color images of $\mathbb{C}_\mathrm{A}$ and $\mathbb{C}_\mathrm{B}$ described in Section II-B2 are presented in Fig. 3.

3) Evaluation Metric: The corners of the check patterns on the average RGB images $\bar{\mathcal{C}}^a$ and $\bar{\mathcal{C}}^b$ are manually labeled in order to establish several common-corner 2D point pairs. Afterwards, an automatic corner refinement approach with sub-pixel accuracy is employed to refine the coordinates of these 2D corner points [34]. Let a common-corner 2D point pair be denoted by $(\boldsymbol{u}_i^a, \boldsymbol{u}_i^b)$ as the description in Section II-B. This 2D point pair is then converted into a 3D point

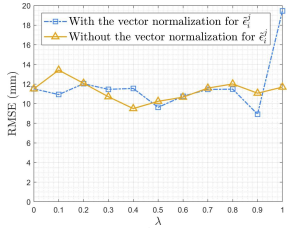


Figure 4. The influence of changing λ in the feature concatenation strategy, *i.e.* (6), on the registration performance of two Kinect V2 cameras.

pair $\left(\boldsymbol{x}_i^a, \boldsymbol{x}_i^b\right)$ by using (3) and $\hat{\mathcal{D}}^j$. Note that, because of the intensity-related distance error [35], [36] of any ToF sensor, the depth value d_i^j for a 2D corner point $\boldsymbol{u}_i^j$ is filtered by a specific filter in [37], where the depth information of only the white checks around $\boldsymbol{u}_i^j$ is taken into account. The Root-Mean-Square Error (RMSE) metric is applied to evaluate the performance of different Kinect V2 registration methods:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left\|\mathbf{T}\boldsymbol{x}_i^a - \boldsymbol{x}_i^b\right\|_2^2}. \tag{7}$$

Here, $n = 20$. All the common-corner 2D point pairs are indicated by green circles in Fig. 3. Besides, the four common-corner 2D point pairs represented by red crosses on a board are utilized to calculate the coarse estimation result in LiveScan3D [18]. Note that this board plays the same role as a marker.

4) Implementation Details: The SURF interest point detector and feature descriptor are implemented by referring to their implementations in OpenCV with default parameters. Each voxel in the volumetric grid of the TDF representation has the same size of $0.01^3\,\mathrm{m}^3$. The pre-trained 3DMatch network from [22] has been optimized on multiple scene reconstruction datasets in diverse real-world environments at varying scales.

B. Results and Analysis

1) Quantitative Evaluation: The varying λ in Section II-B6 for the feature concatenation strategy has different impacts on the performance of the coarse estimation phase as shown in Fig. 4. The yellow solid line stands for the registration precision of changing λ in (6). It can be found that only using SURF descriptor ($\lambda = 0$) and using 3DMatch descriptor alone ($\lambda = 1$) have similar RMSE results ($\approx 11.6\,\mathrm{mm}$), which indicates that both color and geometry descriptors in the coarse estimation stage are effective for the calibration of the two Kinect V2 cameras. Besides, when $\lambda = 0.4$, the best camera registration performance is achieved (RMSE $= 9.497\,\mathrm{mm}$), which implies that the combination of both color and geometry information is beneficial for the camera registration task of Kinect V2 sensors. Since the 3DMatch descriptor $\tilde{\boldsymbol{\epsilon}}_i^j$ is not normalized, a vector normalization method is tried here through dividing $\tilde{\boldsymbol{\epsilon}}_i^j$ by a Euclidean norm before the concatenation operation for the feature descriptors. The blue dash line reveals the performance of feature concatenation using the normalized $\tilde{\boldsymbol{\epsilon}}_i^j$ at varying λ. When using the normalized 3DMatch descriptor alone ($\lambda = 1$), the RMSE value increases dramatically compared with the case of using the original 3DMatch descriptor alone, which suggests that the vector normalization for $\tilde{\boldsymbol{\epsilon}}_i^j$ is not helpful for the registration of the Kinect V2 cameras. Moreover, a reasonable best registration performance is achieved at $\lambda = 0.5$, which demonstrates that both color and geometry descriptors are of equal importance for the coarse rigid transformation estimation again.

Table I
THE RMSE RESULTS OF DIFFERENT METHODS.

Method	Coarse Estimation (mm)	Estimation Refinement (mm)
LiveScan3D [18]	12.714	20.116
Gao *et al.* [21]	79.037	32.416
Proposed	**9.497**	20.221

The performance comparison between the proposed method and baseline approaches is illustrated in Table I. Here, for the proposed method, $\lambda = 0.4$ without 3DMatch descriptor normalization is used for the performance comparison, which is explained by the detailed analysis as above. As can be seen from the table, the proposed Kinect V2 registration method with only using coarse estimation achieves the best performance, which proves the effectiveness of the proposed camera registration method for Kinect V2 sensors using both color and deep geometry information. However, the estimation refinement step does not reduce the RMSE values for LiveScan3D and the proposed method, which means that the ICP-based estimation refinement algorithm may get stuck in a local minimum that can be even worse than an initialization, *i.e.* the coarse estimation result. The estimation refinement step is effective only for method [21], whereas its performance is the worst among these three approaches, which suggests that estimation refinement will be a necessary step if the camera registration error of coarse estimation is large.

2) Qualitative Evaluation: The proposed Kinect V2 registration method is also evaluated qualitatively as illustrated in Fig. 5. Here, for each Kinect V2 camera, an integration algorithm in KinectFusion [38] is adopted to fuse all the depth images $\mathcal{D}_k^j$ into a 3D voxel grid using a volumetric TSDF representation [28]. Specifically, a projective point-to-point distance metric for the voxel-to-surface distance approximation and a constant weighting function are used in this integration algorithm [39]. Afterwards, the marching-cubes algorithm is utilized to extract a mesh standing for the zero-level isosurface encoded by the TSDF representation [40]. In Fig. 5, the yellow mesh comes from $\mathbb{C}_\mathrm{A}$ and it has been transformed into the camera coordinates of $\mathbb{C}_\mathrm{B}$ by using the rigid transformation result, *i.e.* $\mathbf{T}_1$, of the proposed method. The gray mesh is from $\mathbb{C}_\mathrm{B}$. It is apparent that these two meshes coincide very well, which demonstrates that the proposed Kinect V2 registration method using feature concatenation strategy for both SURF and 3DMatch features is effective for the Kinect V2 calibration problem in this static scene.

IV. CONCLUSION

In this paper, a Kinect V2 registration method using color (SURF) and deep geometry (3DMatch) feature descriptors is presented. The proposed method is integrated into a coarse-to-fine framework and it achieves better performance in the

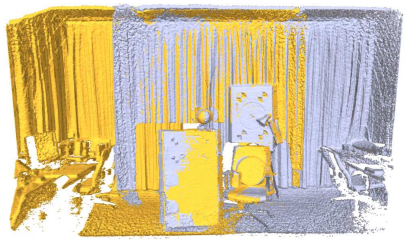


Figure 5. The visualized camera registration result of the proposed method using a TSDF representation. The yellow mesh is from $\mathbb{C}_{\mathrm{A}}$ and the gray mesh is from $\mathbb{C}_{\mathrm{B}}$. Both of them are in the camera space of $\mathbb{C}_{\mathrm{B}}$.

coarse estimation stage than in the estimation refinement phase for a static scene. Moreover, for the proposed method, using the combination of color and geometry features performs better than using color or geometry feature alone. Furthermore, the experimental performance comparison shows the superiority of the proposed method over other baseline approaches.

Acknowledgments

The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, and the German Research Foundation (DFG) No. K02044/8-1.

References

[1] A. Corti, S. Giancola, G. Mainetti, and R. Sala, "A metrological characterization of the Kinect V2 time-of-flight camera," *Robotics and Autonomous Systems*, vol. 75, pp. 584–594, 2016. 1

[2] H. Sarbolandi, D. Lefloch, and A. Kolb, "Kinect range sensing: Structured-light versus time-of-flight Kinect," *CVIU*, vol. 139, pp. 1–20, 2015. 1

[3] O. Wasenmüller and D. Stricker, "Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision," in *ACCV Workshops*, 2016, pp. 34–45. 1, 2

[4] S. Esquivel, Y. Gao, T. Michels, L. Palmieri, and R. Koch, "Synchronized data capture and calibration of a large-field-of-view moving multi-camera light field rig," in *3DTV-CON Workshops*, 2016. 1

[5] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field image processing: An overview," *IEEE J-STSP*, vol. 11, no. 7, pp. 926–954, 2017. 1

[6] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *CVPR*, 2015, pp. 343–352. 1

[7] A. Smolic, "3D video and free viewpoint video - from capture to display," *Pattern Recognition*, vol. 44, no. 9, pp. 1958–1968, 2011. 1

[8] J. Yu, "A light-field journey to virtual reality," *IEEE MultiMedia*, vol. 24, no. 2, pp. 104–112, 2017. 1

[9] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018. 1

[10] Y. Gao and R. Koch, "Parallax view generation for static scenes using parallax-interpolation adaptive separable convolution," in *ICME Workshops*, 2018. 1

[11] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017. 1

[12] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field reconstruction using deep convolutional network on EPI," in *CVPR*, 2017, pp. 1638–1646. 1

[13] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM TOG*, vol. 35, no. 6, pp. 193:1–193:10, 2016. 1

[14] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Image based rendering technique via sparse representation in shearlet domain," in *ICIP*, 2015, pp. 1379–1383. 1

[15] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment - A modern synthesis," in *Vision Algorithms: Theory and Practice*, 2000, pp. 298–372. 1

[16] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE TPAMI*, vol. 14, no. 2, pp. 239–256, 1992. 1

[17] H. Afzal, D. Aouada, D. Font, B. Mirbach, and B. Ottersten, "RGB-D multi-view system calibration for full 3D scene reconstruction," in *ICPR*, 2014, pp. 2459–2464. 1

[18] M. Kowalski, J. Naruniec, and M. Daniluk, "LiveScan3D: A fast and inexpensive 3D data acquisition system for multiple Kinect v2 sensors," in *3DV*, 2015, pp. 318–325. 1, 2, 4

[19] V. Soleimani, M. Mirmehdi, D. Damen, S. Hannuna, and M. Camplani, "3D data acquisition and registration using two opposing kinects," in *3DV*, 2016, pp. 128–137. 1

[20] D.-M. Córdova-Esparza, J. R. Terven, H. Jiménez-Hernández, and A.-M. Herrera-Navarro, "A multiple camera calibration and point cloud fusion tool for Kinect V2," *SCP*, vol. 143, pp. 1–8, 2017. 1

[21] Y. Gao, S. Esquivel, R. Koch, M. Ziegler, F. Zilly, and J. Keinert, "A novel Kinect V2 registration method for large-displacement environments using camera and scene constraints," in *ICIP*, 2017, pp. 997–1001. 1, 4

[22] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser, "3DMatch: Learning local geometric descriptors from RGB-D reconstructions," in *CVPR*, 2017, pp. 199–208. 1, 4

[23] Y. Gao, S. Esquivel, R. Koch, and J. Keinert, "A novel self-calibration method for a stereo-ToF system using a Kinect V2 and two 4K GoPro cameras," in *3DV*, 2017. 2

[24] Y. Gao, M. Ziegler, F. Zilly, S. Esquivel, and R. Koch, "A linear method for recovering the depth of Ultra HD cameras using a Kinect V2 sensor," in *IAPR MVA*, 2017, pp. 494–497. 2

[25] P. H. Schönemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, 1966. 2

[26] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE TPAMI*, vol. PAMI-9, no. 5, pp. 698–700, 1987. 2

[27] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *ECCV*, 2006, pp. 404–417. 3

[28] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *SIGGRAPH*, 1996, pp. 303–312. 3, 4

[29] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Comm. of the ACM*, vol. 18, no. 9, pp. 509–517, 1975. 3

[30] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992. 3

[31] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *JOSA A*, vol. 4, no. 4, pp. 629–642, 1987. 3

[32] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, no. 6, pp. 381–395, 1981. 3

[33] D. C. Brown, "Close-range camera calibration," *Photogrammetric Engineering*, vol. 37, no. 8, pp. 855–866, 1971. 3

[34] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A toolbox for easily calibrating omnidirectional cameras," in *IROS*, 2006, pp. 5695–5701. 3

[35] A. Kolb, E. Barth, R. Koch, and R. Larsen, "Time-of-flight cameras in computer graphics," *CGF*, vol. 29, no. 1, pp. 141–159, 2010. 4

[36] M. Lindner, I. Schiller, A. Kolb, and R. Koch, "Time-of-flight sensor calibration for accurate range sensing," *CVIU*, vol. 114, no. 12, pp. 1318–1328, 2010. 4

[37] V. Garro, C. Dal Mutto, P. Zanuttigh, and G. M. Cortelazzo, "A novel interpolation scheme for range data with side information," in *CVMP*, 2009, pp. 52–60. 4

[38] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *ISMAR*, 2011, pp. 127–136. 4

[39] E. Bylow, J. Sturm, C. Kerl, F. Kahl, and D. Cremers, "Real-time camera tracking and 3D reconstruction using signed distance functions." in *RSS*, 2013. 4

[40] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," in *SIGGRAPH*, vol. 21, no. 4, 1987, pp. 163–169. 4

6.6 Publication 6

MAST: Mask-Accelerated Shearlet Transform for Densely-Sampled Light Field Reconstruction

Yuan Gao, Robert Bregovic, Atanas Gotchev and Reinhard Koch

Published in

2019 IEEE International Conference on Multimedia and Expo (ICME), Shanghai, China, 8-12 July 2019, pages 187-192.

DOI: 10.1109/ICME.2019.00040

MAST: MASK-ACCELERATED SHEARLET TRANSFORM FOR DENSELY-SAMPLED LIGHT FIELD RECONSTRUCTION

Yuan Gao, Robert Bregovic, Atanas Gotchev and Reinhard Koch

Kiel University, Germany
{yga, rk}@informatik.uni-kiel.de

Tampere University, Finland
{robert.bregovic, atanas.gotchev}@tuni.fi

ABSTRACT

Shearlet Transform (ST) is one of the most effective algorithms for the Densely-Sampled Light Field (DSLF) reconstruction from a Sparsely-Sampled Light Field (SSLF) with a large disparity range. However, ST requires a precise estimation of the disparity range of the SSLF in order to design a shearlet system with decent scales and to pre-shear the sparsely-sampled Epipolar-Plane Images (EPIs) of the SSLF. To overcome this limitation, a novel coarse-to-fine DSLF reconstruction method, referred to as Mask-Accelerated Shearlet Transform (MAST), is proposed in this paper. Specifically, a state-of-the-art learning-based optical flow method, FlowNet2, is employed to estimate the disparities of a SSLF. The estimated disparities are then utilized to roughly estimate the densely-sampled EPIs for the sparsely-sampled EPIs of the SSLF. Finally, an elaborately-designed soft mask for a coarsely-inpainted EPI is exploited to perform an iterative refinement on this EPI. Experimental results on nine challenging horizontal-parallax real-world SSLF datasets with large disparity ranges (up to 35 pixels) demonstrate the effectiveness and efficiency of the proposed method over the other state-of-the-art approaches.

***Index Terms*—** View Synthesis, Parallax View Generation, Densely-Sampled Light Field Reconstruction, Shearlet Transform, Mask-Accelerated Shearlet Transform

1. INTRODUCTION

Densely-Sampled Light Field (DSLF) is a discrete representation of the 4D approximation of the plenoptic function parameterized by two parallel planes (camera plane and image plane) [1], where multi-perspective camera views are arranged in such a way that the disparities between adjacent views are less than one pixel [2]. As can be seen in Fig. 1 (a), a horizontal-parallax light field capture system can be considered as a camera moving along the horizontal axis. All the parallax views captured by this camera constitute a ground-truth 3D light field volume as illustrated in Fig. 1 (b). This volume can then be turned into ground-truth Epipolar-Plane Images (EPIs), of which an example is shown in Fig. 1 (c). A Sparsely-Sampled Light Field (SSLF) for this horizontal-parallax light field dataset consists of views with blue borders. The virtual cameras represented by dash-line triangles with yellow color correspond to the target "unknown" views to be reconstructed, the number of which is decided by the interpolation rate δ. A sparsely-sampled EPI ε from the ground-truth EPI ψ is presented in Fig. 1 (d). The DSLF reconstruction for the SSLF can be treated as reconstructing a densely-sampled EPI f from the sparsely-sampled EPI ε. If the ground-truth EPI ψ is not densely sampled, it will be necessary to downsample the reconstructed densely-sampled EPI f to construct a target EPI with the same size as ψ (see Sect. 4.1).

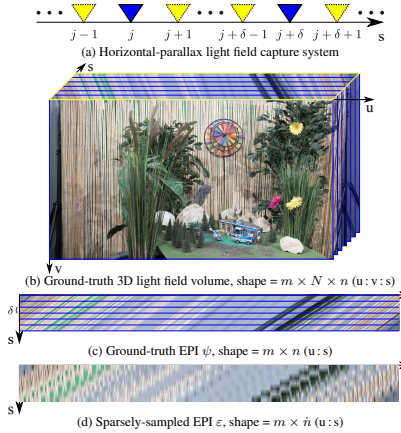


Fig. 1. Introduction to the DSLF reconstruction problem.

Shearlet Transform (ST) [3, 4] is extremely effective in reconstructing a densely-sampled EPI from a sparsely-sampled EPI with a large disparity range. This algorithm typically needs to obtain the disparity range of the sparsely-sampled EPI to construct a specifically-tailored universal shearlet system [3, 5] with decent scales. Besides, the sparsely-sampled EPI also needs the disparity range information for shearing and padding in order to be correctly processed by this elaborately-designed shearlet system. Moreover, for DSLF reconstruction from SSLFs with large disparity ranges, this algorithm is prone to be time-consuming due to the high num-

ber of iterations of its iterative thresholding algorithm. Therefore, in this paper, a novel ST-based coarse-to-fine DSLF reconstruction method, referred to as Mask-Accelerated Shearlet Transform (MAST), is proposed to address these two problems. The presented MAST method takes full advantage of a state-of-the-art learning-based optical flow estimation approach, *i.e.* FlowNet2 [6], to estimate the disparities of the whole SSLF for resolving the first problem. In addition, the estimated disparities are also used to roughly restore a densely-sampled EPI from a sparsely-sampled EPI via inverse warping. The iterative estimation refinement algorithm in ST convergences faster by means of an elaborately-designed soft mask for the coarsely-inpainted densely-sampled EPI, thus tackling the second problem. Experimental results demonstrate the superior performance of MAST over the other state-of-the-art DSLF reconstruction methods on nine challenging horizontal-parallax real-world light field datasets with disparity ranges up to 35 pixels.

2. RELATED WORK

High-quality and high-fidelity Virtual Reality (VR) [7] and Free Viewpoint Video (FVV) [8] contents fundamentally rely on DSLFs for the reason that DSLFs can be turned into continuous light fields via linear interpolation [9]. However, due to the difficulty of directly capturing a DSLF, a DSLF is typically reconstructed from a SSLF. The challenging DSLF reconstruction problem has been tried to be solved by light field angular super-resolution-based approaches, most of which treat it as novel view synthesis problem and do not consider the disparity range of the input SSLF. Kalantari *et al*. propose a learning-based approach composed of a disparity estimator and a color predictor to synthesize novel views from four corner sub-aperture views of a micro-lens array-based light field camera [10]. Wu *et al*. utilize a residual-learning method to restore the angular detail of EPIs within a blur-deblur framework [11]. However, the maximum disparity of the SSLF that can be handled by this approach is only 5 pixels. Yeung *et al*. also design a learning-based view synthesis network consisting of view synthesis and refinement components to reconstruct DSLFs [12]. Nevertheless, for different interpolation rates, their network needs to be retrained. Gao and Koch utilize a state-of-the-art video frame interpolation method, *i.e*. adaptive Separable Convolution (SepConv) [13], and a fine-tuning strategy enhancing the convolution kernels of SepConv to reconstruct DSLFs in a recursive way [14].

3. METHODOLOGY

3.1. DSLF reconstruction using ST

The shearlet transform approach for DSLF reconstruction is originally proposed in [3] and extended in [4] with computational acceleration. Given a coarsely-sampled EPI $\varepsilon \in \mathbb{R}^{m\times\dot{n}}$ from a SSLF as shown in Fig. 1 (d), ST reconstructs a desired densely-sample EPI $f \in \mathbb{R}^{m\times\ddot{n}}$ by an iterative inpainting algorithm using the sparse representation of f in shearlet domain. The sampling interval of the desired EPI f for rearranging the rows of the input decimated EPI ε is denoted by τ as illustrated in Fig. 2 (a). Since the desired EPI f to be reconstructed is densely sampled, it is apparent that $\tau \geqslant d_{range}$ and d_{range} stands for the disparity range of the input decimated EPI ε, *i.e.* $d_{range} = d_{max} - d_{min}$. It should be noted that a pre-shearing process relying on d_{min} is typically necessary for the input decimated EPI ε in order to make sure that the new $d'_{min} = 0$ and $d'_{max} = d_{range}$. Besides, the vertical sizes of the input decimated EPI ε and reconstructed densely-sampled EPI f meet the condition that $\ddot{n} = (\dot{n} - 1)\tau + 1$.

The reconstruction of the desired densely-sampled EPI f is typically performed via an iterative inpainting process with t iterations, corresponding to the intermediate reconstructed EPI result f_i, $i \in [1, t] \cap \mathbb{Z}$. Besides, the shearlet analysis transform for reconstructing f is defined as $S : \mathbb{R}^{m\times\ddot{n}} \rightarrow \mathbb{R}^{\eta\times m\times\ddot{n}}$ and the shearlet synthesis transform is denoted by $S^* : \mathbb{R}^{\eta\times m\times\ddot{n}} \rightarrow \mathbb{R}^{m\times\ddot{n}}$. Additionally, f_0 stands for the coarse estimation of f, which is a zero-padded EPI for the input decimated EPI ε, *i.e.* $f_0(:\tau:,:) = \varepsilon$ as shown in Fig. 2 (a). The reconstruction of f_i during iteration i is performed using the double relaxation method [4], which has been demonstrated to be faster and more robust than the original hard-thresholding algorithm in [3]:

$$\begin{aligned} \hat{f}_i &= S^*\Big(T_{\lambda_i}\big(S\big(f_i + \alpha(f_0 - M \circ f_i)\big)\big)\Big)\,, \\ \tilde{f}_i &= \hat{f}_i + \beta_1(\hat{f}_i - f_{i-1})\,, \\ f_{i+1} &= \tilde{f}_i + \beta_2(\tilde{f}_i - f_{i-2})\,, \end{aligned} \tag{1}$$

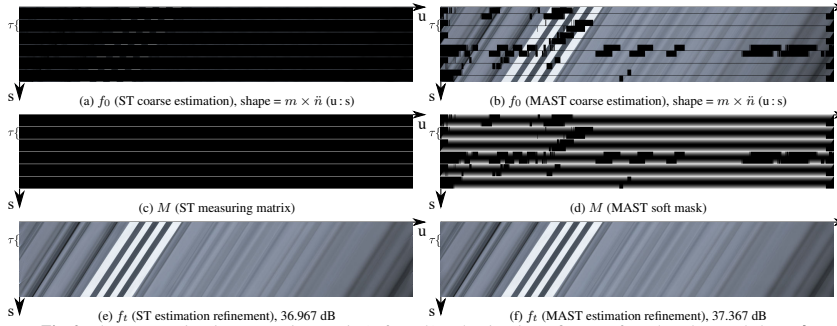
where

$$\begin{aligned} \beta_1 &= \frac{\mathrm{sum}\big((f_0 - \hat{f}_i) \circ M \circ (\hat{f}_i - f_{i-1})\big)}{\mathrm{sum}\big((\hat{f}_i - f_{i-1}) \circ M \circ (\hat{f}_i - f_{i-1})\big)}\,, \\ \beta_2 &= \frac{\mathrm{sum}\big((f_0 - \tilde{f}_i) \circ M \circ (\tilde{f}_i - f_{i-2})\big)}{\mathrm{sum}\big((\tilde{f}_i - f_{i-2}) \circ M \circ (\tilde{f}_i - f_{i-2})\big)}\,. \end{aligned} \tag{2}$$

Here, $\mathrm{sum}(\cdot)$ returns the sum of all the elements in the input matrix, α is a parameter for adjusting the convergence speed, '$\circ$' denotes the element-wise (Hadamard) product and M is a logical measuring matrix as shown in Fig. 2 (c), where ideally $f_t \circ M = f_0$. In addition, $T_{\lambda_i}(\cdot)$ is a hard-thresholding operator [15] for the threshold value λ_i, which linearly decreases from λ_{max} to λ_{min} with iteration i increasing from 1 to t. As can be seen from (1) and (2), the computation time of the ST approach above is linearly dependent on the maximum iteration number t. A reliable f_0, *i.e.* coarse estimation of f, makes it feasible to accelerate ST with a smaller t.

3.2. Mask-Accelerated Shearlet Transform (MAST)

In order to make a more reliable estimation of f_0 *w.r.t.* the desired densely-sampled EPI f, one of the state-of-the-art learning-based optical flow methods, *i.e.* FlowNet2 [6], is utilized to estimate bidirectional flow between adjacent views in a horizontal-parallax SSLF, $\mathcal{D}^{\mathrm{sslf}} = \{\mathcal{I}_i | 1 \leqslant i \leqslant \dot{n}\}$, of which the corresponding unknown DSLF is denoted by $\mathcal{D}^{\mathrm{dslf}} = \{\tilde{\mathcal{I}}_r | 1 \leqslant r \leqslant \ddot{n}\}$. Since a horizontal-parallax SSLF does not have vertical motions of image objects between any two neighboring views, only the horizontal component of the



(a) f_0 (ST coarse estimation), shape = $m \times \ddot{n}$ (u : s)
(b) f_0 (MAST coarse estimation), shape = $m \times \ddot{n}$ (u : s)
(c) M (ST measuring matrix)
(d) M (MAST soft mask)
(e) f_t (ST estimation refinement), 36.967 dB
(f) f_t (MAST estimation refinement), 37.367 dB

Fig. 2. The coarse estimation, measuring matrix (soft mask) and estimation refinement for a densely-sampled EPI f.

optical flow displacement vector is kept after the bidirectional flow estimation. The bidirectional flow between $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$ in the $\mathcal{D}^{\text{sslf}}$ is represented by $F_{i\to i+1}$ and $F_{i+1\to i}$. A forward-backward consistency constraint [16] between $F_{i\to i+1}$ and $F_{i+1\to i}$ is applied here to roughly remove the inaccuracies caused by occlusions and large motions of image objects. Let $\dot{r} = (r-1)\%\tau$[1] and $i = 1 + (r - \dot{r} - 1)/\tau$, the estimated bidirectional flow is then used to perform a coarse estimation of the parallax images in $\mathcal{D}^{\text{dslf}}$ as follows[2]:

$$\tilde{\mathcal{I}}_r = \begin{cases} \mathcal{I}_i & \text{for} \quad \dot{r} = 0\,, \\ g(\mathcal{I}_i,\, -\frac{\dot{r}}{\tau} F_{i\to i+1}) & \text{for} \quad 0 < \dot{r} < \frac{\tau}{2}\,, \\ g(\mathcal{I}_{i+1},\, -\frac{(\tau-\dot{r})}{\tau} F_{i+1\to i}) & \text{for} \quad \frac{\tau}{2} < \dot{r} < \tau\,, \\ \mathbf{0} & \text{for} \quad \dot{r} = \frac{\tau}{2}\,. \end{cases} \tag{3}$$

Here, $g(\cdot,\,\cdot)$ is an inverse warping function using bicubic interpolation [17]. The roughly-estimated $\mathcal{D}^{\text{dslf}}$ is then turned into densely-sampled EPIs, such that the coarse estimation f_0 of f is partially restored as displayed in Fig. 2 (b). Note that the large missing areas are caused by the filtering of the unreliable optical flows using the bidirectional consistency check. However, the roughly inpainted areas in f_0 are not accurate enough for directly using ST. Specifically, due to the accumulation error of the optical flow in the interpolation algorithm in (3), horizontal lines of f_0 near the locations, *i.e.* $\dot{r} = \frac{\tau}{2}$, have larger inpainting errors than those near the ground-truth regions, *i.e.* $\dot{r} = 0$. Therefore, a novel ST-based method, Mask-Accelerated Shearlet Transform (MAST), is proposed to solve this problem by replacing the measuring matrix in (1) and (2) with an elaborately-designed soft mask, *i.e.*

$$M(r,\,c) = \begin{cases} 1.0 & \text{for} \quad \dot{r} = 0\,, \\ \omega(1 - \frac{2\dot{r}}{\tau})^2 & \text{for} \quad \dot{r} > 0\,,\ f_0(r,\,c) > 0\,, \\ 0 & \text{for} \quad \dot{r} > 0\,,\ f_0(r,\,c) = 0\,, \end{cases} \tag{4}$$

where $\omega \in (0,\,1)$, $r \in [1,\,\ddot{n}] \cap \mathbb{Z}$ and $c \in [1,\,m] \cap \mathbb{Z}$. An example soft mask corresponding to f_0 is illustrated in Fig. 2 (d). It can be seen that this mask suppresses the contributions of f_0 in the regions which are not inpainted or meet the condition that $\dot{r}$ is close to $\frac{\tau}{2}$; however, it enhances the contributions from the ground-truth nearby areas, thus effectively improving the initialization of the densely-sampled EPIs for the iterative double relaxation-based ST in Sect. 3.1.

[1]Here, '%' stands for the modulo operation.

[2]Assume that $\tau\%2 = 0$ for this $\mathcal{D}^{\text{dslf}}$.

4. EXPERIMENTS

4.1. Experimental settings

Datasets. The high density camera array dataset [18] is a real-world 4D light field dataset that can be utilized to evaluate light field angular super-resolution methods with large disparity ranges. Nine different scenes in this dataset are captured by a high-resolution and high-definition DSLR camera in a precise gantry system, such that nine corresponding light field sub-datasets are built. Eight of these sub-datasets have an angular resolution of 101×21. The remaining one has an angular resolution of 99×21. The spatial resolution of all the sub-datasets is 3976×2652 pixels. The raw images in each sub-dataset have black areas near the image borders, which is due to the calibration, and large disparities between neighboring views, which make it difficult to use these raw images as ground-truth light field data directly. To overcome this limitation, a cutting and scaling strategy is proposed as shown in Fig. 3 (j). In particular, a bottom-right $16:9$ image is cut from a raw image with preserving 95% of the width of this raw image. The cut image is then resized to 1024×576 pixels using bicubic interpolation. Finally, only the top 97 images after the process of the cutting and scaling strategy for each sub-dataset are kept and used as the ground-truth horizontal-parallax light field dataset $\mathcal{D}_\mu$, $\mu \in [1,\,9] \cap \mathbb{Z}$. In other words, $\mathcal{D}_\mu = \{\mathcal{I}_j^\mu | 1 \leqslant j \leqslant n\}$, $\mathcal{I}_j^\mu \in \mathbb{R}^{m\times N}$, where $n = 97$, $m = 1024$ and $N = 576$. The middle image, *i.e.* $\mathcal{I}_{49}^\mu$, of each ground-truth 3D light field dataset $\mathcal{D}_\mu$ is exhibited in Fig. 3 (a)-(i). The SSLF $\mathcal{D}_\mu^{\text{sslf}}$ from $\mathcal{D}_\mu$ is generated by using an interpolation rate $\delta\ (= 16)$ as shown in Fig. 1 (a) and (c), such that $\mathcal{D}_\mu^{\text{sslf}} = \{\mathcal{I}_i^\mu | 1 \leqslant i \leqslant \dot{n}\}$, $\dot{n} = (n-1)/\delta + 1$.

(a) $\mathcal{D}_1$: Books and charts (b) $\mathcal{D}_2$: Lego city (c) $\mathcal{D}_3$: Lightfield production (d) $\mathcal{D}_4$: Plants (e) $\mathcal{D}_5$: Table in the garden

(f) $\mathcal{D}_6$: Table top I (g) $\mathcal{D}_7$: Table top II (h) $\mathcal{D}_8$: Table top III (i) $\mathcal{D}_9$: Workshop (j) Cutting and scaling

Fig. 3. The middle views of nine evaluation datasets and (j) illustrates the image cutting and scaling strategy in Sect. 4.1.

Table I. The minimum and average per-view PSNR results (in dB, explained in Sect. 4.1) for the performance evaluation of different DSLF reconstruction methods on nine light field evaluation datasets.

Minimum per-view PSNR value (dB) of DSLF reconstruction on $\mathcal{D}_\mu$									
Method	$\mathcal{D}_1$	$\mathcal{D}_2$	$\mathcal{D}_3$	$\mathcal{D}_4$	$\mathcal{D}_5$	$\mathcal{D}_6$	$\mathcal{D}_7$	$\mathcal{D}_8$	$\mathcal{D}_9$
SepConv ($\mathcal{L}_1$) [13]	23.324	20.341	23.912	25.059	27.080	28.344	20.419	21.208	26.369
PIASC ($\mathcal{L}_1$) [14]	23.311	20.343	23.915	25.065	27.092	28.396	20.416	21.208	26.377
ST [4]	28.881	22.725	26.252	27.718	29.418	**32.485**	**23.186**	23.518	28.710
MAST	**30.167**	**22.965**	**26.866**	**27.920**	**29.541**	32.448	23.119	**23.847**	**29.001**
Average per-view PSNR value (dB) of DSLF reconstruction on $\mathcal{D}_\mu$									
Method	$\mathcal{D}_1$	$\mathcal{D}_2$	$\mathcal{D}_3$	$\mathcal{D}_4$	$\mathcal{D}_5$	$\mathcal{D}_6$	$\mathcal{D}_7$	$\mathcal{D}_8$	$\mathcal{D}_9$
SepConv ($\mathcal{L}_1$) [13]	26.220	22.569	26.251	27.645	28.719	29.868	22.929	23.500	28.546
PIASC ($\mathcal{L}_1$) [14]	26.231	22.587	26.281	27.697	28.777	29.921	22.941	23.529	28.595
ST [4]	30.122	24.107	28.294	**29.487**	30.358	33.361	**24.431**	**25.417**	30.605
MAST	**31.286**	**24.214**	**28.740**	29.356	**30.371**	**33.768**	24.226	25.390	**30.624**

Table II. The average computation time of reconstructing a densely-sampled EPI (RGB channels) using ST and MAST.

Average computation time (s)		
Method	$\tau = 32$	$\tau = 48$
ST [4]	7.966	14.867
MAST	**2.813**	**5.073**

The DSLF to be reconstructed is $\mathcal{D}_\mu^{\text{dslf}} = \{\tilde{\mathcal{I}}_r^\mu | 1 \leqslant r \leqslant \ddot{n}\}$, $\ddot{n} = (\dot{n} - 1)\tau + 1$ as described in Sect. 3.

Disparity estimation. The horizontal disparities between neighboring views in each $\mathcal{D}_\mu^{\text{sslf}}$ are calculated via the optical flow algorithm in Sect. 3.2. The estimated minimum disparity d_{min}, maximum disparity d_{max} and disparity range d_{range} are illustrated in Fig. 5. The sampling interval τ should be as small as possible in order to save computation time for both ST and MAST, while it has two constraints that $\tau \% \delta = 0$ and $\tau \geqslant d_{range}$ (see Sect. 3.1). Therefore, it can be seen from the figure that the best sampling interval τ for datasets $\mathcal{D}_\mu$, $\mu \in \{1, 2, 7\}$ is 32 and for the other six datasets, $\tau = 48$.

Evaluation criteria. The per-view PSNR for a ground-truth dataset $\mathcal{D}_\mu$ and the reconstructed $\mathcal{D}_\mu^{\text{dslf}}$ from $\mathcal{D}_\mu^{\text{sslf}}$ for it is described as below:

$$\text{MSE}_j^\mu = \frac{1}{3 \cdot m \cdot N} \sum_{x=1}^{m} \sum_{y=1}^{N} \left\| \tilde{\mathcal{I}}^\mu_{\frac{\tau(j-1)}{\delta}+1}(x, y) - \mathcal{I}_j^\mu(x, y) \right\|_2^2 ,$$

$$\text{PSNR}_j^\mu = 10 \log_{10} \left(\frac{255^2}{\text{MSE}_j^\mu} \right). \tag{5}$$

The minimum and average per-view PSNRs constitute the evaluation criteria for the evaluation of different DSLF reconstruction methods on a dataset $\mathcal{D}_\mu$.

Implementation details. For a dataset $\mathcal{D}_\mu$, the construction of a specifically-designed universal shearlet system [3] with ξ scales relies on the sampling interval τ of it, *i.e.* $\xi = \lceil \log_2 \tau \rceil$. The parameter ω in (4) is set to 0.1. The maximum threshold value λ_{max} and minimum threshold value λ_{min} are set to 8 and 0.04, respectively. Note that these two values are for the case of using a normalized coarsely-sampled EPI ε, *i.e.*

$$\varepsilon = \frac{\varepsilon - \min(\varepsilon)}{\max(\varepsilon) - \min(\varepsilon)} , \tag{6}$$

where $\max(\cdot)$ and $\min(\cdot)$ return the maximum value and the minimum value of an input matrix, respectively. The reconstructed f using this normalized ε is then rescaled back to original range of values via

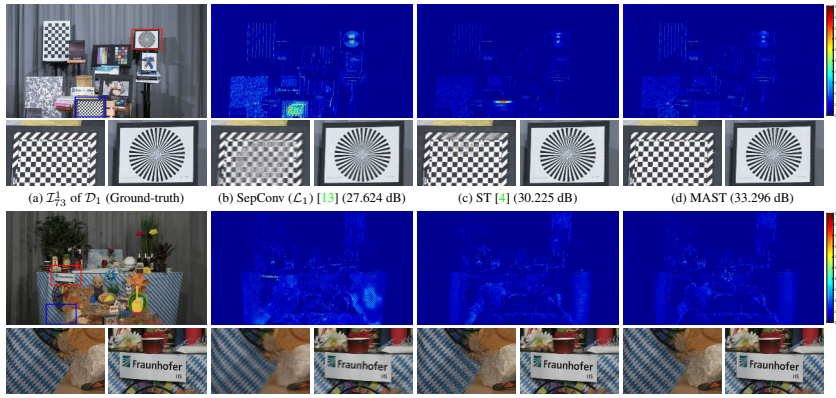
$$f = \big(\max(\varepsilon) - \min(\varepsilon)\big) f + \min(\varepsilon) . \tag{7}$$

Besides, for the maximum iteration number of ST, $t = 100$ and for that of MAST, $t = 30$. Regarding the parameter controlling the convergence speed in (1), $\alpha = 30$. Both ST and MAST are implemented by using CUDA and executed on an Nvidia GeForce GTX Titan X 12 GB GPU.

4.2. Results and analysis

The proposed method and baseline approaches are evaluated quantitatively and qualitatively as follows:

Quantitative evaluation. The minimum and average per-view PSNR values of using different DSLF reconstruction methods on different horizontal-parallax light field datasets



(a) $\mathcal{I}^1_{73}$ of $\mathcal{D}_1$ (Ground-truth) (b) SepConv ($\mathcal{L}_1$) [13] (27.624 dB) (c) ST [4] (30.225 dB) (d) MAST (33.296 dB)

(e) $\mathcal{I}^6_{25}$ of $\mathcal{D}_6$ (Ground-truth) (f) SepConv ($\mathcal{L}_1$) [13] (28.863 dB) (g) ST [4] (33.326 dB) (h) MAST (34.024 dB)

Fig. 4. The visualization of the DSLF-reconstruction quality of using different methods.

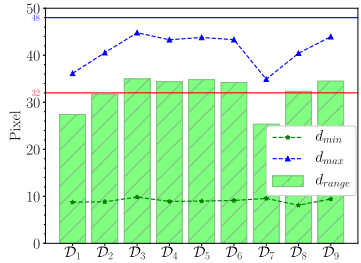


Fig. 5. The disparity estimations of $\mathcal{D}_\mu$.

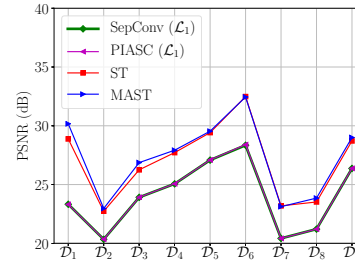


Fig. 6. The minimum per-view PSNR results on $\mathcal{D}_\mu$.

are presented in Table I. It can be seen from the minimum per-view PSNR data that the proposed MAST method achieves the best performance on most of the datasets except for $\mathcal{D}_6$ and $\mathcal{D}_7$. However, on these two datasets, the minimum per-view PSNR values of ST are only 0.037 dB and 0.067 dB higher than those of MAST. With regard to the average per-view PSNR data at the bottom of Table I, MAST still outperforms the other baseline methods on most datasets except for $\mathcal{D}_4$, $\mathcal{D}_7$ and $\mathcal{D}_8$, which demonstrates the effectiveness of the proposed DSLF reconstruction approach again. The computation efficiency of both ST and MAST is evaluated in terms of computation time as shown in Table II. The proposed MAST is significantly faster than ST, *i.e.* MAST requires only $\approx 35\%$ computation time of ST, which is mainly because MAST has 30 refinement iterations while ST needs 100 iterations for the DSLF reconstruction on the challenging horizontal-parallax real-world light field datasets. This also demonstrates that MAST is much more efficient than ST for DSLF reconstruction. Note that the computation time of optical flow estimation and inverse warping parts of MAST can be ignored compared with the computation time of the iterative EPI-refinement process since both of them are performed in real-time.

Qualitative evaluation. The minimum per-view PSNR data for all the DSLF reconstruction methods on different light field evaluation datasets are plotted in Fig. 6. It is apparent that both SepConv and PIASC have almost the same minimum per-view PSNR results on all the datasets, which are much lower than ST and MAST. This suggests that the two DSLF reconstruction methods using the state-of-the-art video frame interpolation technology are not appropriate for DSLF reconstruction from SSLFs with large disparity ranges. Besides, the proposed MAST approach outperforms ST on most of the challenging light field datasets, which indicates that MAST is more effective than ST for DSLF reconstruction. The reconstructed images using different DSLF reconstruc-

tion methods are visualized and compared in Fig. 4. Since SepConv and PIASC have similar DSLF reconstruction performance, only SepConv is compared here. For the top row, the image parts of the checkerboard and Siemens star on $\mathcal{I}_{73}^{1}$ of $\mathcal{D}_1$ are chosen as the interesting areas to be compared. It can be seen from Fig. 4 (b) that the reconstructed checkerboard using SepConv has serious blur artifacts, which is mainly because the the size of repetitive check patterns of the checkerboard is much smaller than the disparities of it, such that SepConv is incapable of knowing the true motion of this checkerboard. As can be seen from Fig. 4 (c), the recovered checkerboard using ST is slightly better than that of using SepConv, while the reconstructed Siemens star has obvious artifacts. In Fig. 4 (d), the proposed MAST method achieves the best reconstruction performance with visually correct and sharp results, which proves the effectiveness of the proposed MAST method composed of optical-flow-based coarse estimation and mask-assisted iterative estimation refinement for EPIs. Regarding the bottom row Fig. 4, part of the tablecloth with foreground and the Fraunhofer IIS logo are selected as the interesting areas from $\mathcal{I}_{25}^{6}$ of $\mathcal{D}_6$. Both of the reconstructed results in Fig. 4 (f) using SepConv are blur, which, on the one hand, is caused by the small size of the repetitive pattern of the tablecloth; on the other hand, the size of the convolution kernels of SepConv is only 51×51, restricting the performance of it in handling DSLF reconstruction from SSLFs with large disparity ranges. The DSLF reconstruction results of ST in Fig. 4 (g) do not have this kind of "blur" problem. However, the reconstructed tablecloth area has evident artifacts, which are well handled by the proposed MAST method as illustrated in Fig. 4 (h). It implies that the optical-flow-based coarse estimation and mask-assisted iterative estimation refinement in MAST are beneficial to improving the final DSLF reconstruction performance.

5. CONCLUSION

This paper presents a novel coarse-to-fine method, MAST, for DSLF reconstruction from SSLFs with large disparity ranges. The proposed MAST method fully leverages a state-of-the-art optical flow estimation method, *i.e.* FlowNet2, to roughly estimate a densely-sampled EPI from a sparsely-sampled EPI. Based on the coarsely-inpainted densely-sampled EPI and the inevitable error accumulation of any optical flow algorithm, a soft mask is elaborately designed for the iterative hard-thresholding-based estimation refinement approach in ST. Experimental results show that MAST achieves better performance than the other state-of-the-art DSLF reconstruction methods on nine challenging real-world horizontal-parallax light field datasets with large disparity ranges (up to 35 pixels). Moreover, MAST is a time-efficient algorithm that is nearly three times faster than ST.

Acknowledgments. The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, and the German Research Foundation (DFG) No. K02044/8-1. We thank Nvidia for their GPU donation.

6. REFERENCES

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*, 1996, pp. 31–42. 1

[2] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Image based rendering technique via sparse representation in shearlet domain," in *ICIP*, 2015, pp. 1379–1383. 1

[3] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018. 1, 2, 4

[4] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017. 1, 2, 4, 5

[5] M. Genzel and G. Kutyniok, "Asymptotic analysis of inpainting via universal shearlet systems," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 2301–2339, 2014. 1

[6] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "FlowNet 2.0: Evolution of optical flow estimation with deep networks," in *CVPR*, 2017, pp. 1647–1655. 2

[7] J. Yu, "A light-field journey to virtual reality," *IEEE MultiMedia*, vol. 24, no. 2, pp. 104–112, 2017. 2

[8] A. Smolic, "3D video and free viewpoint video - from capture to display," *Pattern Recognition*, vol. 44, no. 9, pp. 1958–1968, 2011. 2

[9] Z. Lin and H.-Y. Shum, "A geometric analysis of light field rendering," *IJCV*, vol. 58, no. 2, pp. 121–138, 2004. 2

[10] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM TOG*, vol. 35, no. 6, pp. 193:1–193:10, 2016. 2

[11] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field reconstruction using deep convolutional network on EPI," in *CVPR*, 2017, pp. 1638–1646. 2

[12] H.W.F. Yeung, J. Hou, J. Chen, Y.Y. Chung, and X. Chen, "Fast light field reconstruction with deep coarse-to-fine modelling of spatial-angular clues," in *ECCV*, 2018, pp. 138–154. 2

[13] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *ICCV*, 2017, pp. 261–270. 2, 4, 5

[14] Y. Gao and R. Koch, "Parallax view generation for static scenes using parallax-interpolation adaptive separable convolution," in *ICME Workshops*, 2018, pp. 1–4. 2, 4

[15] H. Lakshman, W.-Q. Lim, H. Schwarz, D. Marpe, G. Kutyniok, and T. Wiegand, "Image interpolation using shearlet based sparsity priors," in *ICIP*, 2013, pp. 655–659. 2

[16] J. Hur and S. Roth, "MirrorFlow: Exploiting symmetries in joint optical flow and occlusion estimation," in *ICCV*, 2017, pp. 312–321. 3

[17] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super SloMo: High quality estimation of multiple intermediate frames for video interpolation," in *CVPR*, 2018, pp. 9000–9008. 3

[18] M. Ziegler, R. op het Veld, J. Keinert, and F. Zilly, "Acquisition system for dense lightfield of large scenes," in *3DTV-CON*, 2017, pp. 1–4. 3

6.7 Publication 7

Light Field Reconstruction Using Shearlet Transform in TensorFlow

Yuan Gao, Reinhard Koch, Robert Bregovic and Atanas Gotchev

Published in

2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Shanghai, China, 8-12 July 2019, pages 612-612.

DOI: 10.1109/ICMEW.2019.00116

LIGHT FIELD RECONSTRUCTION USING SHEARLET TRANSFORM IN TENSORFLOW

Yuan Gao, Reinhard Koch

Kiel University, Germany
{yga, rk}@informatik.uni-kiel.de

Robert Bregovic, Atanas Gotchev

Tampere University, Finland
{robert.bregovic, atanas.gotchev}@tuni.fi

ABSTRACT

Shearlet Transform (ST) is one of the most effective approaches for light field reconstruction from Sparsely-Sampled Light Fields (SSLFs). This demo paper presents a comprehensive implementation of ST for light field reconstruction using one of the most popular machine learning libraries, *i.e.* TensorFlow. The flexible architecture of TensorFlow allows for the easy deployment of ST across different platforms (CPUs, GPUs, TPUs) running varying operating systems with high efficiency and accuracy.

***Index Terms*—** Light Field Reconstruction, Shearlet Transform, TensorFlow, Epipolar-Plane Image, Light Field Sparsification

1. INTRODUCTION

Shearlet Transform (ST) [1, 2] is designed for reconstructing a Densely-Sampled Light Field (DSLF) from a Sparsely-Sampled Light Field (SSLF) using Epipolar-Plane Image (EPI) sparse representation in shearlet domain. Typically, ST is composed of four steps, which are (1) pre-shearing, (2) shearlet system construction, (3) sparsity regularization and (4) post-shearing. For step (1), (2) and (4), ST requires the information of minimum disparity d_{min}, maximum disparity d_{max} and disparity range d_{range} of the input SSLF, so that this input SSLF can be pre-sheared with new $d'_{min} = 0$ and $d'_{max} = d_{range}$. Besides, a shearlet system for this input SSLF can be constructed with ξ scales, where $\xi = \lceil \log_2 d_{range} \rceil$. Regarding sparsity regularization, it typically consists of analysis transform, hard thresholding and synthesis transform as introduced in [1]. In addition, the double overrelaxation (DORE) algorithm in [2] can efficiently accelerate the convergence speed of sparsity regularization.

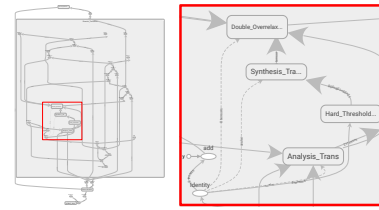
2. IMPLEMENTATION

The sparsity regularization step of ST is re-implemented here using TensorFow as shown in Fig. 1 (a). The original implementation [2] of ST using Matlab and the presented TensorFlow ST implementation are also compared here. For a fair comparison, the fifth row of "Dishes" in 4D Light Field Dataset [3] is extracted and decimated with an interpolation rate $\delta = 2$. In other words, the input SSLF contains 5 horizontal-parallax images. Since the "Dishes" light field has ground-truth d_{min} (-3.1 pixels) and d_{max} (3.5 pixels), the disparity condition of the input SSLF can be derived. The pre-shearing step is performed as illustrated in Fig. 1 (b) and (c). An Nvidia Titan Xp is exploited to process these 512 sparsely-sampled EPIs with resolution of 608×128 pixels and the number of iterations is set to 30.

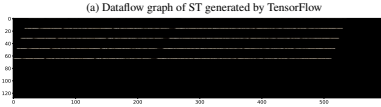
Table I. Computation time comparison of different implementations.

ST implementation	Time (s)
Matlab (with CUDA) [2]	87.635
TensorFlow (with CUDA)	91.574

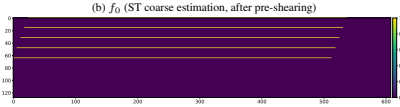
The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, and the German Research Foundation (DFG) No. K02044/8-1. The Titan Xp used for this research was donated by the NVIDIA Corporation.



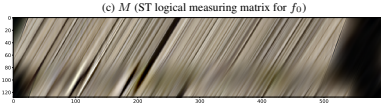
(a) Dataflow graph of ST generated by TensorFlow



(b) f_0 (ST coarse estimation, after pre-shearing)



(c) M (ST logical measuring matrix for f_0)



(d) f_{30} (ST estimation refinement, after 30 iterations)

Fig. 1. The coarse estimation, logical measuring matrix and estimation refinement for densely-sampled EPI reconstruction using ST.

The computation time of different implementations are compared in Table I. As can be seen from it, the presented TensorFlow implementation achieves comparable performance to the original Matlab implementation. An example of a reconstructed densely-sampled EPI is displayed in Fig. 1 (d). The source code of this demo will be released to facilitate learning-based light field research using ST.

3. REFERENCES

[1] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018.

[2] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017.

[3] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4D light fields," in *ACCV*, 2016, pp. 19–34.

6.8 Publication 8

IEST: Interpolation-Enhanced Shearlet Transform for Light Field Reconstruction Using Adaptive Separable Convolution

Yuan Gao, Reinhard Koch, Robert Bregovic and Atanas Gotchev

Published in

2019 27th European Signal Processing Conference (EUSIPCO), A Coruña, Spain, 2-6, Sept. 2019, pages 1-5.

DOI: 10.23919/EUSIPCO.2019.8903168

IEST: Interpolation-Enhanced Shearlet Transform for Light Field Reconstruction Using Adaptive Separable Convolution

Yuan Gao, Reinhard Koch
Department of Computer Science
Kiel University
24118 Kiel, Germany
{yga, rk}@informatik.uni-kiel.de

Robert Bregovic, Atanas Gotchev
Faculty of Information Technology and Communication Sciences
Tampere University
33014 Tampere, Finland
{robert.bregovic, atanas.gotchev}@tuni.fi

***Abstract*—The performance of a light field reconstruction algorithm is typically affected by the disparity range of the input Sparsely-Sampled Light Field (SSLF). This paper finds that (i) one of the state-of-the-art video frame interpolation methods, *i.e.* adaptive Separable Convolution (SepConv), is especially effective for the light field reconstruction on a SSLF with a small disparity range (< 10 pixels); (ii) one of the state-of-the-art light field reconstruction methods, *i.e.* Shearlet Transformation (ST), is especially effective in reconstructing a light field from a SSLF with a moderate disparity range (10-20 pixels) or a large disparity range (> 20 pixels). Therefore, to make full use of both methods to solve the challenging light field reconstruction problem on SSLFs with moderate and large disparity ranges, a novel method, referred to as Interpolation-Enhanced Shearlet Transform (IEST), is proposed by incorporating these two approaches in a coarse-to-fine manner. Specifically, ST is employed to give a coarse estimation for the target light field, which is then refined by SepConv to improve the reconstruction quality of parallax views involving small disparity ranges. Experimental results show that IEST outperforms the other state-of-the-art light field reconstruction methods on nine challenging horizontal-parallax evaluation SSLF datasets of different real-world scenes with moderate and large disparity ranges.**

***Index Terms*—Light Field Reconstruction, Parallax View Generation, Adaptive Separable Convolution, Shearlet Transform, Interpolation-Enhanced Shearlet Transform**

I. Introduction

A light field can be considered as a 4D approximation of the plenoptic function parameterized by two parallel planes (camera plane and image plane) [1]; therefore, a 4D light field is typically composed of camera images sampled on a regular 2D grid [2] or an irregular 2D grid [3]. If the disparities between adjacent views in a light field are less than one pixel, this light field can be referred to as a Densely-Sampled Light Field (DSLF) [4]. How to capture a horizontal-parallax light field is illustrated in Fig. 1. As can be seen from this figure, the horizontal-parallax desired light field is captured by a system with cameras uniformly distributed along the horizontal axis 's' with the same camera orientation. Let this desired target light field be denoted by $\mathcal{D} = \{\mathcal{I}_i | 1 \leqslant i \leqslant m\}$. Due to the hardware limitations of most of the light field capture systems in real-world environments, it is difficult for them to capture all the m parallax images of the desired target light field $\mathcal{D}$ with small disparities. In other words, such camera systems can capture only part of parallax views of the target light field $\mathcal{D}$, which are represented by the solid-line green cameras in Fig. 1. Let this Sparsely-Sampled Light Field (SSLF) be denoted by $\mathcal{S}$, $\mathcal{S} \subseteq \mathcal{D}$ and $|\mathcal{S}| = n$ $(< m)$. This paper aims to solve the problem of reconstructing the missing parallax views for the target light field $\mathcal{D}$ from the input SSLF $\mathcal{S}$. The relationship between them is determined by the interpolation rate δ, where $\delta = \frac{m-1}{n-1}$. It is obvious that different interpolation rates correspond to different disparity conditions for the input SSLF $\mathcal{S}$. Besides, if the target light field $\mathcal{D}$ is a DSLF, the light field reconstruction on $\mathcal{S}$ can be called DSLF reconstruction.

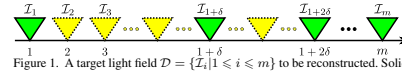


Figure 1. A target light field $\mathcal{D} = \{\mathcal{I}_i | 1 \leqslant i \leqslant m\}$ to be reconstructed. Solid-line green triangles constitute an input SSLF $\mathcal{S}$. Dash-line yellow triangles are the missing parallax views to be reconstructed for the target light field $\mathcal{D}$.

Motivation. The adaptive Separable Convolution (SepConv) approach [5] is one of the state-of-the-art video frame interpolation methods, which is extended by Gao and Koch in [6] for solving the DSLF reconstruction problem in a recursive manner, treating an input horizontal-parallax light field as a video captured by a virtual camera moving horizontally. The restriction of SepConv is that it may fail in light field reconstruction on SSLFs with larger disparity ranges, because its novel view synthesis ability is limited by the size of the convolution kernels. For more details refer to Sect. IV-B. Alternatively, one of the state-of-the-art light field reconstruction methods, *i.e.* Shearlet Transform (ST) [7, 8], is a universal solution to DSLF reconstruction and does not suffer from such restriction. This paper focuses on investigating how to employ the advantages of these two methods to better reconstruct light fields from SSLFs with moderate and large disparity ranges.

To address the challenging light field reconstruction problem for the cases of moderate and large disparity ranges, a novel method, referred to as Interpolation-Enhanced Shearlet Transform (IEST), is proposed in this paper. The proposed IEST method fully leverages the advantages of both ST and

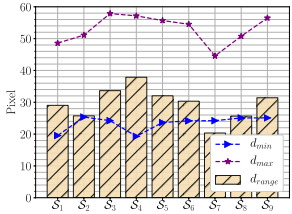
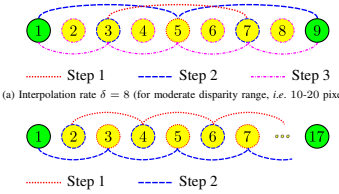


Figure 2. Disparity estimation for nine evaluation SSLFs $\mathcal{S}_\mu$ with an interpolation rate $\delta = 16$ using PWC-Net [9].

SepConv in a coarse-to-fine manner to reconstruct a target light field from a horizontal-parallax SSLF with a moderate or large disparity range. Specifically, ST is applied to reconstruct the target light field $\mathcal{D}$ from an input SSLF $\mathcal{S}$, so that the missing parallax views in $\mathcal{S}$ are coarsely estimated. Two elaborately-designed parallax view refinement strategies, corresponding to different interpolation rates $\delta \in \{8, 16\}$, are then performed on the coarsely-estimated $\mathcal{D}$ in a recursive way. Experimental results indicate that IEST outperforms all the other state-of-the-art methods on nine challenging horizontal-parallax evaluation SSLF datasets for both the moderate disparity range (10-19 pixels) and the large disparity range (20-38 pixels). Moreover, for any evaluation SSLF dataset with a small disparity range (5-9.5 pixels), SepConv achieves better light field reconstruction performance than ST.

II. Related Work

Learning-based video frame synthesis. Niklaus *et al*. employ a deep fully Convolutional Neural Network (CNN) to estimate pixel-wise spatially-adaptive 2D convolution kernels, which are applied on the two consecutive input video frames to synthesize an intermediate one [10]. However, for each image pixel, this method predicts a $n \times n$ ($n = 41$) convolution kernel, which will be prohibitively expensive in memory requirement if the input images are in high resolution. To tackle this problem, Niklaus *et al*. propose a spatially-adaptive Separable Convolution (SepConv) approach, which approximates each of the 2D convolution kernels with a pair of 1D kernels, thus reducing the number of kernel parameters from n^2 to $2n$ for each 2D convolution kernel [5]. Liu *et al*. propose an end-to-end deep network, *i.e*. Deep Voxel Flow (DVF), to synthesize a video frame in either interpolation or extrapolation with sharp results [11]. More recently, Niklaus *et al*. fully leverage a state-of-the-art optical flow algorithm, *i.e*. PWC-Net [9], to estimate bidirectional flow between two consecutive input video frames, which is applied to pre-warp the input video frames together with their corresponding per-pixel context maps extracted by a pre-trained neural network [12]. All these pieces of pre-warped information are then fed to a video frame synthesis network, *i.e*. a modified GridNet [13], to interpolate an intermediate video frame at a desired temporal position. Jiang *et al*. also estimate bidirectional optical flow between two consecutive input video frames via a flow computation CNN [14]. The estimated optical flow is then refined by a flow interpolation



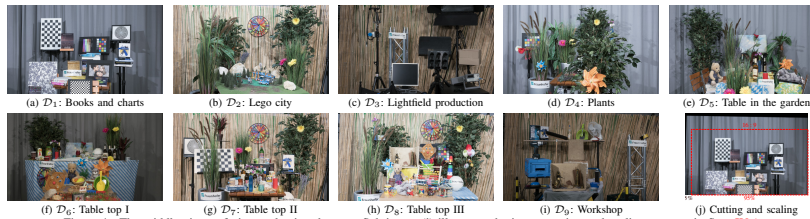
(a) Interpolation rate $\delta = 8$ (for moderate disparity range, *i.e*. 10-20 pixels)

(b) Interpolation rate $\delta = 16$ (for large disparity range, *i.e*. > 20 pixels)

Figure 3. Flowcharts of IEST for light field reconstruction from SSLFs at different interpolation rates, *i.e*. $\delta \in \{8, 16\}$.

CNN, which additionally predicts a soft visibility map. Both the refined optical flow and predicted soft visibility map are utilized to interpolate an intermediate video frame at any arbitrary time step via warping and fusion. Meyer *et al*. apply the steerable pyramid filters [15] to decompose the input two consecutive video frames [16]. Their decompositions, consisting of amplitudes, phases and low-pass residuals, are fed to a decoder-only neural network, *i.e*. PhaseNet, to predict the corresponding decomposition of the intermediate frame in order to fulfill image reconstruction. Visually preferable results are achieved by this method in challenging scenarios containing lighting changes or motion blur.

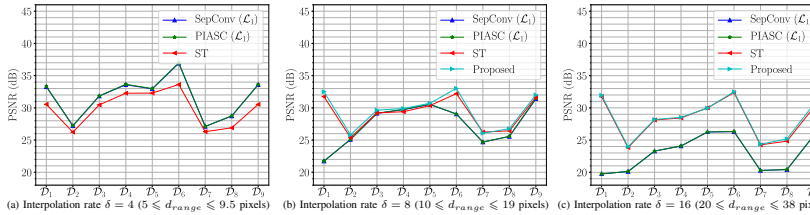
Light field angular super-resolution. Kalantari *et al*. propose a learning-based view synthesis approach, which is composed of disparity and color estimators, for synthesizing novel views from a sparse set of sub-aperture images of a micro-lens array-based consumer light field camera [17]. Wu *et al*. present a blur-restoration-deblur framework for Epipolar-Plane Image (EPI) interpolation to reconstruct dense light fields [18]. A residual network with three convolution layers is utilized to restore the angular detail of a blurred and up-sampled EPI. However, due to the limitation in the blurring kernel size and bicubic interpolation, this method can only handle SSLF data with very small disparity ranges (up to 5 pixels). Vagharshakyan *et al*. reconstruct DSLFs from SSLFs by exploiting EPI sparsification in shearlet domain, which has been demonstrated to be effective in reconstructing Lambertian scenes and non-Lambertian scenes containing semi-transparent objects [7, 8]. Gao and Koch employ a fine-tuning strategy to enhance the motion-sensible convolution kernels of the state-of-the-art video frame interpolation method, *i.e*. SepConv, and propose Parallax-Interpolation Adaptive Separable Convolution (PIASC) to reconstruct a DSLF from a horizontal-parallax SSLF in a recursive way [6]. Yeung *et al*. design an end-to-end 4D convolutional light field reconstruction network consisting of view synthesis and view refinement phases for fast light field reconstruction from a SSLF [19]. Wang *et al*. also propose an end-to-end learning framework for fast light field reconstruction [20]. Their network includes two 2D strided convolutions for the interpolation of stacked sparsely-sampled EPIs and two detail-restoration 3D CNNs for restoring high-frequency details of these interpolated EPI volumes. In conclusion, studies in [19, 20] can only be applied on full-



(a) $\mathcal{D}_1$: Books and charts (b) $\mathcal{D}_2$: Lego city (c) $\mathcal{D}_3$: Lightfield production (d) $\mathcal{D}_4$: Plants (e) $\mathcal{D}_5$: Table in the garden

(f) $\mathcal{D}_6$: Table top I (g) $\mathcal{D}_7$: Table top II (h) $\mathcal{D}_8$: Table top III (i) $\mathcal{D}_9$: Workshop (j) Cutting and scaling

Figure 4. The middle views of nine evaluation datasets. Sub-image (j) illustrates the image cutting and scaling strategy in Sect. IV-A.



(a) Interpolation rate $\delta = 4$ ($5 \leqslant d_{range} \leqslant 9.5$ pixels) (b) Interpolation rate $\delta = 8$ ($10 \leqslant d_{range} \leqslant 19$ pixels) (c) Interpolation rate $\delta = 16$ ($20 \leqslant d_{range} \leqslant 38$ pixels)

Figure 5. Minimum per-view PSNR results (in dB, explained in Sect. IV-A) of different light field reconstruction methods on nine evaluation datasets with different interpolation rates $\delta \in \{4, 8, 16\}$.

parallax SSLF data. In addition, for different intermediate-view interpolation factors, *i.e.* δ in Sect. I, these methods need to re-train their networks. Nevertheless, this paper focuses on investigating a universal light field reconstruction solution *w.r.t.* input SSLFs with moderate and large disparity ranges.

III. Methodology

A. Shearlet Transform (ST)

For tackling the DSLF reconstruction problem on SSLFs with varying disparities, ST is originally proposed in [7] and extended in [8, 21, 22]. The key idea of ST is to design an elaborately-tailored universal shearlet system [7, 23], which is exploited to perform sparsity regularization in the shearlet transform domain for the sparsely-sampled EPIs of an input SSLF via an iterative α-adaptive algorithm [7] or a double overrelaxation (DORE) algorithm [8]. The performance of the ST algorithm relies on the precision of the disparity estimation of the input SSLF. Specifically, the minimum disparity d_{min} and maximum disparity d_{max} of this SSLF should be precisely estimated before applying ST. The corresponding disparity range of the input SSLF is derived from d_{min} and d_{max}, *i.e.* $d_{range} = (d_{max} - d_{min})$. Based on the value of the estimated d_{min}, a pre-shearing step using cubic interpolation is then applied on the input SSLF, so that the new minimum disparity $d'_{min} = 0$, the new maximum disparity $d'_{max} = d_{range}$ and the sheared input SSLF is able to be effectively and efficiently processed by a shearlet system with ξ scales, where $\xi = \lceil \log_2 d_{range} \rceil$. Finally, a post-processing shearing procedure is performed on the reconstructed DSLF in order to compensate for the loss of the minimum disparity that is eliminated in the pre-shearing step.

B. Interpolation-Enhanced Shearlet Transform (IEST)

Although ST is a universal solution to the light field reconstruction problem on SSLFs with varying disparities, it is not as effective as one of the state-of-the-art video frame interpolation methods, *i.e.* SepConv, for light field reconstruction from SSLFs with small disparity ranges. An example for this phenomenon is shown in Fig. 5 (a), where the interpolation rate $\delta = 4$ equals to $5 \leqslant d_{range} \leqslant 9.5$ pixels, derived from $20 \leqslant d_{range} \leqslant 38$ pixels in the case of $\delta = 16$ in Fig. 2, for all the evaluation SSLFs $\mathcal{S}_\mu$. However, for SSLFs with moderate and large disparity ranges, ST tends to be more effective than SepConv as illustrated in Fig. 5 (b) and (c). Intuitively, taking advantage of SepConv to refine the parallax views of light fields that are reconstructed by ST from SSLFs with moderate and large disparity ranges may improve the final light field reconstruction performance. Therefore, a novel light field reconstruction method, *i.e.* Interpolation-Enhanced Shearlet Transform (IEST), is proposed. The IEST method is specially designed for light field reconstruction on SSLFs with moderate and large disparity ranges with a consideration that the reconstructed parallax views of ST involving small disparity ranges can be refined by SepConv. Depending on different interpolation rates, two parallax view refinement strategies of IEST are presented in Fig. 3. As shown in (a), the first strategy is designed for the case of interpolation rate $\delta = 8$. Here, green circles stand for the ground-truth parallax views from an input SSLF $\mathcal{S}$ (also see Fig. 1) and yellow circles denote the parallax views reconstructed by ST. The reconstructed parallax views represented by yellow circles are then refined by SepConv recursively, which is depicted by using three types of dash lines that represent Step 1, 2 and 3.

Table I
MINIMUM AND AVERAGE PER-VIEW PSNR RESULTS (IN DB, EXPLAINED IN SECT. IV-A) FOR THE PERFORMANCE EVALUATION OF DIFFERENT LIGHT FIELD RECONSTRUCTION METHODS ON NINE EVALUATION DATASETS.

Minimum PSNR
(Interpolation rate $\delta = 8$ and $10 \leqslant d_{range} \leqslant 19$ pixels)

	SepConv ($\mathcal{L}_1$) [5]	PIASC ($\mathcal{L}_1$) [6]	ST [8]	Proposed
$\mathcal{D}_1$	21.733	21.731	31.750	**32.505**
$\mathcal{D}_2$	25.087	25.103	25.375	**25.807**
$\mathcal{D}_3$	29.145	29.161	29.220	**29.644**
$\mathcal{D}_4$	29.729	29.760	29.399	**29.893**
$\mathcal{D}_5$	30.525	30.557	30.349	**30.713**
$\mathcal{D}_6$	29.039	29.044	32.203	**33.023**
$\mathcal{D}_7$	24.685	24.688	**26.237**	26.067
$\mathcal{D}_8$	25.558	25.576	26.438	**26.763**
$\mathcal{D}_9$	31.379	31.466	31.644	**32.005**

Minimum PSNR
(Interpolation rate $\delta = 16$ and $20 \leqslant d_{range} \leqslant 38$ pixels)

	SepConv ($\mathcal{L}_1$) [5]	PIASC ($\mathcal{L}_1$) [6]	ST [8]	Proposed
$\mathcal{D}_1$	19.753	19.742	31.754	**31.941**
$\mathcal{D}_2$	20.118	20.123	23.839	**23.964**
$\mathcal{D}_3$	23.289	23.295	28.125	**28.194**
$\mathcal{D}_4$	24.073	24.084	28.430	**28.529**
$\mathcal{D}_5$	26.254	26.262	**30.004**	29.955
$\mathcal{D}_6$	26.307	26.317	32.368	**32.504**
$\mathcal{D}_7$	20.283	20.283	24.257	**24.350**
$\mathcal{D}_8$	20.419	20.423	24.831	**25.184**
$\mathcal{D}_9$	25.628	25.639	30.021	**30.498**

Average PSNR
(Interpolation rate $\delta = 8$ and $10 \leqslant d_{range} \leqslant 19$ pixels)

	SepConv ($\mathcal{L}_1$) [5]	PIASC ($\mathcal{L}_1$) [6]	ST [8]	Proposed
$\mathcal{D}_1$	24.498	24.495	33.429	**33.826**
$\mathcal{D}_2$	26.625	26.648	26.666	**27.144**
$\mathcal{D}_3$	30.781	30.828	30.875	**31.534**
$\mathcal{D}_4$	32.344	32.446	32.189	**32.972**
$\mathcal{D}_5$	31.762	31.840	32.328	**32.613**
$\mathcal{D}_6$	31.333	31.383	34.843	**36.034**
$\mathcal{D}_7$	27.209	27.228	27.383	**27.827**
$\mathcal{D}_8$	27.299	27.332	27.732	**28.151**
$\mathcal{D}_9$	32.878	32.951	32.836	**33.407**

Average PSNR
(Interpolation rate $\delta = 16$ and $20 \leqslant d_{range} \leqslant 38$ pixels)

	SepConv ($\mathcal{L}_1$) [5]	PIASC ($\mathcal{L}_1$) [6]	ST [8]	Proposed
$\mathcal{D}_1$	21.514	21.505	33.220	**33.435**
$\mathcal{D}_2$	22.779	22.807	25.285	**25.467**
$\mathcal{D}_3$	26.009	26.044	29.764	**29.833**
$\mathcal{D}_4$	27.408	27.483	30.677	**30.933**
$\mathcal{D}_5$	28.005	28.063	31.337	**31.457**
$\mathcal{D}_6$	28.779	28.838	34.100	**34.224**
$\mathcal{D}_7$	23.375	23.397	25.915	**25.992**
$\mathcal{D}_8$	23.158	23.196	26.542	**26.775**
$\mathcal{D}_9$	28.416	28.478	32.083	**32.328**

To be precise, each step uses two parallax views having a small disparity range to synthesize the middle view between them. For the interpolation rate $\delta = 16$, the second strategy of IEST, as shown in (b), refines the reconstructed parallax views from ST with only two steps. It is worth to be mentioned that the strategy of IEST designed for $\delta = 8$ is especially effective for the light field reconstruction on SSLFs with moderate disparity ranges (10-20 pixels), while the second IEST strategy designed for $\delta = 16$ is more effective for the light field reconstruction on SSLFs with large disparity ranges (> 20 pixels).

IV. EXPERIMENTS

A. Experimental Settings

Evaluation datasets. The dataset of High Density Camera Array (HDCA) [24] is used for evaluating the performance of all methods. This dataset has nine high-fidelity dense light fields for different real-world scenes captured by a movable high-resolution and high-quality DSLR camera. Eight of them have the same angular resolution of 101×21. The remaining one has an angular resolution of 99×21. The spatial resolution of these light fields is 3976×2652 pixels. Since the proposed method and baseline approaches are originally designed for parallax view generation using horizontal-parallax SSLFs, only the top 97 horizontal-parallax views of each light field are selected for evaluation. However, these raw images have a problem that some boundary regions have no color information due to calibration, which is not fair for performance comparison of different methods. To overcome this limitation, an image cutting and scaling strategy is proposed as illustrated Fig. 4 (j). In particular, a 95%-width image (at the right of the original raw image) is cut and a $16:9$-shape image at the bottom of this cut image is then downscaled to a new resolution of 1280×720 pixels using bicubic interpolation. After performing these two operations for all the light fields, nine horizontal-parallax light field datasets $\mathcal{D}_\mu$ are constructed and their middle views are exhibited in Fig. 4 (a)-(i). Note that $m = 97$ for each ground-truth light field dataset $\mathcal{D}_\mu$, where $1 \leqslant \mu \leqslant 9$.

Disparity estimation. From these nine ground-truth light field datasets $\mathcal{D}_\mu$, the corresponding input SSLFs $\mathcal{S}_\mu$ are constructed by using different interpolation rates δ, *i.e.* $\delta \in \{4, 8, 16\}$, as introduced in Sect. I. In order to perform ST method correctly, the disparity conditions of different SSLFs $\mathcal{S}_\mu$ should be estimated precisely. To tackle this problem, a state-of-the-art optical flow method, *i.e.* PWC-Net [9], is applied to estimate the bidirectional flow between neighboring views in $\mathcal{S}_\mu$ for the case of $\delta = 16$. Note that only the horizontal components of the calculated optical flow contain useful information, which are leveraged to compute d_{min}, d_{max} and d_{range} for each $\mathcal{S}_\mu$ as illustrated in Fig. 2. It can be found that the minimum d_{range} for all the SSLFs $\mathcal{S}_\mu$ with the interpolation rate of 16 is around 20 pixels, which suggests that all the target light fields $\mathcal{D}_\mu$ are not DSLFs.

Evaluation criteria. The per-view PSNR is exploited to evaluate the performance of different light field reconstruction methods. Additionally, for any parallax view generation method evaluated on a dataset $\mathcal{D}_\mu$, the minimum and average per-view PSNR values constitute the final evaluation criteria.

Implementation details. All the methods mentioned in this paper are implemented using CUDA and executed on an NVIDIA Titan Xp GPU. The pre-trained neural networks of SepConv and PWC-Net are from [5] and [9], respectively. Besides, the parameters of ST using the DORE algorithm are set as same as [8], where $\alpha = 20$ with 100 iterations and a low-pass initial estimation. The construction of the shearlet system used by ST relies on the estimated disparity range of the input SSLF $\mathcal{S}_\mu$, as explained in Sect. III-A.

B. Results and Analysis

The minimum per-view PSNR values using different light field reconstruction methods on all the evaluation SSLFs $\mathcal{S}_\mu$ at varying interpolation rates, *i.e.* $\delta \in \{4, 8, 16\}$, are presented in Fig. 5. Comparing (a), (b) and (c) in this figure, it can be seen that SepConv achieves better performance than ST on all the evaluation DSLFs for the case that interpolation rate

$\delta = 4$. However, for higher interpolation rates, *i.e.* $\delta \in \{8, 16\}$, the performance of SepConv is significantly worse than that of ST for reconstructing the target light fields $\mathcal{D}_\mu$ from $\mathcal{S}_\mu$. The main reason for this is that (i) SepConv is not capable of correctly interpolating novel views with repetitive patterns that are smaller than the disparities between the input neighboring parallax views, *e.g.*, checkers of the checkerboards in Fig. 4 (a); (ii) the size of the convolution kernel of SepConv is 51×51 pixels, which restricts its novel view synthesis ability *w.r.t.* two parallax images with moderate or large disparities. Besides, directly increasing the convolutional kernel size of SepConv involves re-training the whole network of SepConv and increasing the memory demand, which will not be the best solution to the light field reconstruction problem.

The minimum and average per-view PSNR values of different light field reconstruction methods for interpolation rates $\delta \in \{8, 16\}$ are also shown in Table I. The top row of this table presents the light field reconstruction results for the case of $\delta = 8$, corresponding to $10 \leqslant d_{range} \leqslant 19$ pixels. It can be found that (i) for the minimum-PSNR evaluation criteria, the proposed IEST method achieves the best performance on most of the evaluation DSLFs except for $\mathcal{D}_7$; (ii) for the average-PSNR evaluation criteria, IEST performs significantly better than all the baseline approaches. In addition, on $\mathcal{D}_6$, IEST yields a substantial performance gain of 0.82 and 1.191 dB *w.r.t.* minimum and average PSNRs over the second-best method, *i.e.* ST. This indicates that the proposed IEST method is effective in light field reconstruction on SSLFs with moderate disparity ranges, *e.g.*, up to 19 pixels in given examples. Moreover, SepConv and PIASC have almost the same performance, which is better than ST on $\mathcal{D}_\mu$, $\mu \in \{4, 5\}$ *w.r.t.* minimum-PSNR criterion. The bottom row of Table I shows the light field reconstruction results for the case of $\delta = 16$, corresponding to $20 \leqslant d_{range} \leqslant 38$ pixels. The proposed method outperforms all the other baseline methods on all the evaluation DSLFs except for $\mathcal{D}_5$, where the minimum PSNR of IEST is only 0.049 dB less than that of ST. This suggests that the proposed IEST method is also effective for reconstructing light fields from SSLFs with large disparity ranges, *e.g.*, up to 38 pixels in given examples.

V. Conclusion

In this paper, a novel light reconstruction method, *i.e.* IEST, is presented for reconstructing target light fields from input SSLFs with moderate and large disparity ranges. IEST takes full advantage of a state-of-the-art DSLF reconstruction method, *i.e.* ST, and a state-of-the-art video frame interpolation method, *i.e.* SepConv, to perform light field angular super-resolution in a coarse-to-fine manner. Specifically, SepConv is utilized to refine the light field reconstruction results of ST in a recursive way. Experimental results on nine challenging evaluation datasets demonstrate the effectiveness of IEST over the other state-of-the-art light field reconstruction approaches for both the moderate disparity range (10-19 pixels) and the large disparity range (20-38 pixels).

Acknowledgments. The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, and the German Research Foundation (DFG) No. K02044/8-1. The Titan Xp used for this research was donated by the NVIDIA Corporation.

References

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*, 1996, pp. 31–42.

[2] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," in *ACM TOG*, 2005, vol. 24, pp. 765–776.

[3] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar, "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines," *ACM TOG*, vol. 38, no. 4, pp. 29:1–29:14, 2019.

[4] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Image based rendering technique via sparse representation in shearlet domain," in *ICIP*, 2015, pp. 1379–1383.

[5] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *ICCV*, 2017, pp. 261–270.

[6] Y. Gao and R. Koch, "Parallax view generation for static scenes using parallax-interpolation adaptive separable convolution," in *ICME Workshops*, 2018.

[7] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018.

[8] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017.

[9] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *CVPR*, 2018, pp. 8934–8943.

[10] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *CVPR*, 2017, pp. 2270–2279.

[11] Z. Liu, R.A. Yeh, X. Tang, Y. Liu, and A. Agarwala, "Video frame synthesis using deep voxel flow," in *ICCV*, 2017, pp. 4473–4481.

[12] S. Niklaus and F. Liu, "Context-aware synthesis for video frame interpolation," in *CVPR*, 2018, pp. 1701–1710.

[13] D. Fourure, R. Emonet, E. Fromont, D. Muselet, A. Trémeau, and C. Wolf, "Residual Conv-Deconv grid network for semantic segmentation," in *BMVC*, 2017.

[14] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super SloMo: High quality estimation of multiple intermediate frames for video interpolation," in *CVPR*, 2018, pp. 9000–9008.

[15] E.P. Simoncelli and W.T. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *ICIP*, 1995, vol. 3, pp. 444–447.

[16] S. Meyer, A. Djelouah, B. McWilliams, A. Sorkine-Hornung, M. Gross, and C. Schroers, "PhaseNet for video frame interpolation," in *CVPR*, 2018, pp. 498–507.

[17] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM TOG*, vol. 35, no. 6, pp. 193:1–193:10, 2016.

[18] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field reconstruction using deep convolutional network on EPI," in *CVPR*, 2017, pp. 1638–1646.

[19] H.W.F. Yeung, J. Hou, J. Chen, Y.Y. Chung, and X. Chen, "Fast light field reconstruction with deep coarse-to-fine modelling of spatial-angular clues," in *ECCV*, 2018.

[20] Y. Wang, F. Liu, Z. Wang, G. Hou, Z. Sun, and T. Tan, "End-to-end view synthesis for light field imaging with pseudo 4DCNN," in *ECCV*, 2018, pp. 340–355.

[21] Y. Gao, R. Bregovic, A. Gotchev, and R. Koch, "MAST: Mask-accelerated shearlet transform for densely-sampled light field reconstruction," in *ICME*, 2019.

[22] Y. Gao, R. Koch, R. Bregovic, and A. Gotchev, "FAST: Flow-assisted shearlet transform for densely-sampled light field reconstruction," in *ICIP*, 2019.

[23] M. Genzel and G. Kutyniok, "Asymptotic analysis of inpainting via universal shearlet systems," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 2301–2339, 2014.

[24] M. Ziegler, R. op het Veld, J. Keinert, and F. Zilly, "Acquisition system for dense lightfield of large scenes," in *3DTV-CON*, 2017, pp. 1–4.

6.9 Publication 9

FAST: Flow-Assisted Shearlet Transform for Densely-Sampled Light Field Reconstruction

Yuan Gao, Reinhard Koch, Robert Bregovic and Atanas Gotchev

Published in

2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22-25 Sept. 2019, pages 3741-3745.

DOI: 10.1109/ICIP.2019.8803436

FAST: FLOW-ASSISTED SHEARLET TRANSFORM FOR DENSELY-SAMPLED LIGHT FIELD RECONSTRUCTION

Yuan Gao, Reinhard Koch

Kiel University, Germany
{yga, rk}@informatik.uni-kiel.de

Robert Bregovic, Atanas Gotchev

Tampere University, Finland
{robert.bregovic, atanas.gotchev}@tuni.fi

ABSTRACT

Shearlet Transform (ST) is one of the most effective methods for Densely-Sampled Light Field (DSLF) reconstruction from a Sparsely-Sampled Light Field (SSLF). However, ST requires a precise disparity estimation of the SSLF. To this end, in this paper a state-of-the-art optical flow method, *i.e.* PWC-Net, is employed to estimate bidirectional disparity maps between neighboring views in the SSLF. Moreover, to take full advantage of optical flow and ST for DSLF reconstruction, a novel learning-based method, referred to as Flow-Assisted Shearlet Transform (FAST), is proposed in this paper. Specifically, FAST consists of two deep convolutional neural networks, *i.e.* disparity refinement network and view synthesis network, which fully leverage the disparity information to synthesize novel views via warping and blending and to improve the novel view synthesis performance of ST. Experimental results demonstrate the superiority of the proposed FAST method over the other state-of-the-art DSLF reconstruction methods on nine challenging real-world SSLF sub-datasets with large disparity ranges (up to 26 pixels).

***Index Terms*—** Densely-Sampled Light Field Reconstruction, Parallax View Generation, Novel View Synthesis, Shearlet Transform, Flow-Assisted Shearlet Transform

1. INTRODUCTION

Densely-Sampled Light Field (DSLF) is a discrete representation of the 4D approximation of the plenoptic function parameterized by two parallel planes (camera plane and image plane) [1], where multi-perspective camera views are arranged in such a way that the disparities between adjacent views are less than one pixel [2]. How to reconstruct a DSLF from a Sparsely-Sampled Light Field (SSLF) is depicted in Fig. 1. The solid-line orange cameras, *i.e.* $\mathcal{C}_j$, are uniformly distributed along the horizontal axis with the same camera orientation and focal length. The images captured by them for a static scene compose a horizontal-parallax SSLF. The DSLF reconstruction on this 3D SSLF can be considered as novel view synthesis between any two neighboring parallax views in this SSLF, of which the results are represented by the images from the dash-line blue cameras in Fig. 1. Let the interpolation rate of this novel view synthesis process be denoted by δ, it is apparent that the virtual camera $\mathcal{C}_{j+t}$ meets the condition that $t \in \left\{\frac{1}{\delta}, \frac{2}{\delta}, ..., \frac{\delta-1}{\delta}\right\}$. Besides, in order to reconstruct the target unknown DSLF correctly, δ should be greater than the disparity range of the input SSLF (see Sect. 4.1).

Shearlet Transform (ST) [3, 4] is especially effective in reconstructing a DSLF from a SSLF with a large disparity range (> 16 pixels). The disparity information of the SSLF is required to be obtained in advance for 1) constructing a decent shearlet system; 2) pre-shearing the SSLF in order to eliminate the minimum disparity of it. To tackle the disparity-estimation problem, a state-of-the-art optical flow algorithm, *i.e.* PWC-Net [5], is exploited for estimating the bidirectional disparity maps between adjacent views in the SSLF.

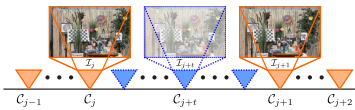


Fig. 1. DSLF reconstruction on a SSLF.

In addition to assisting the DSLF reconstruction of using ST, the estimated bidirectional disparity maps can also be used to perform DSLF reconstruction via novel view synthesis using image warping and blending techniques. However, due to the occlusion and errors in the estimated disparity maps, this disparity-based solution to DSLF reconstruction may not produce visually pleasing results. Therefore, to improve the performance of both ST-based and disparity-based DSLF reconstruction methods, a novel learning-based approach, referred to as Flow-Assisted Shearlet Transform (FAST), is proposed in this paper. The FAST method makes full use of the bidirectional disparity maps predicted by PWC-Net and the DSLF recovered by ST to better reconstruct the target DSLF via two deep convolutional neural networks, *i.e.* disparity refinement network and view synthesis network. Additionally, the proposed FAST is fully convolutional and end-to-end trainable. Experimental results on nine evaluation DSLF sub-datasets demonstrate the effectiveness of FAST for reconstructing DSLFs from SSLFs with large disparity ranges.

2. RELATED WORK

Learning-based novel view synthesis. Niklaus *et al.* interpolate a novel frame between two consecutive video frames using a deep fully Convolutional Neural Network (CNN), which estimates a spatially-adaptive convolution kernel that captures both motion and re-sampling coefficients for each pixel [6]. Due to the large memory demand of outputting all the 2D kernels for all the image pixels, a spatially-adaptive Separable Convolution (SepConv) method is proposed by replacing a 2D kernel with two 1D kernels, thus effectively reducing the memory requirement [7]. Jiang *et al.* propose a CNN-based variable-length multi-frame interpolation method, consisting of a flow computation CNN and a flow interpolation CNN, for generating as many intermediate frames as needed between two consecutive video frames [8].

Light field angular super-resolution. Since a horizontal-parallax SSLF can be converted into a video captured by a virtual camera moving horizontally, Gao and Koch extend SepConv with enhanced kernels and propose Parallax-Interpolation Adaptive Separable Convolution (PIASC) for DSLF reconstruction [9]. In addition, a horizontal-parallax SSLF can also be represented by several Epipolar-Plane Images (EPIs), thus the DSLF reconstruction problem is equal to how to reconstruct densely-sampled EPIs from sparsely-sampled EPIs, which is effectively solved by ST in the shearlet transform domain via EPI sparse representation [10, 3, 4].

3. METHODOLOGY

3.1. Shearlet Transform (ST)

The ST approach is originally proposed in [3] and extended in [4] for addressing the DSLF reconstruction problem with varying disparities. The core idea of ST is to design an elaborately-tailored universal shearlet system [3, 11] and to perform regularization in the shearlet transform domain for EPIs via an iterative α-adaptive algorithm [3] or a double overrelaxation (DORE) algorithm [4]. The construction of the specifically-designed shearlet system relies on the precise disparity estimation of the input SSLF. In particular, the minimum disparity d_{min} and maximum disparity d_{max} of this SSLF are required to be estimated before applying the ST method. The corresponding disparity range of the input SSLF is determined by them, *i.e.* $d_{range} = d_{max} - d_{min}$. Based on the value of the estimated d_{min}, a pre-shearing process using cubic interpolation is then performed on the input SSLF, so that the new minimum disparity $d'_{min} = 0$, the new maximum disparity $d'_{max} = d_{range}$ and the sheared SSLF is capable of being correctly processed by a shearlet system with ξ scales, where $\xi = \lceil \log_2 d_{range} \rceil$. Finally, a post-processing shearing operation is applied to the reconstructed DSLF in order to compensate for the disparity shift that is caused by the pre-shearing step.

3.2. Image warping and blending using optical flow

The bidirectional optical flows between two video frames are effective in interpolating novels views between them via warping and blending [8]. Since a horizontal-parallax SSLF can be treated as a video captured by a virtual camera moving along the horizontal axis, optical flow is used here to solve the DSLF reconstruction problem. Assume $j = 0$ as illustrated in Fig. 1, the novel view $\mathcal{I}_t$ of $\mathcal{C}_t$ between $\mathcal{C}_0$ and $\mathcal{C}_1$ is synthesized by

$$\begin{aligned} \mathcal{I}_t &= \lambda \circ g(\mathcal{I}_0, \mathcal{F}_{t\to 0}) + (1-\lambda) \circ g(\mathcal{I}_1, \mathcal{F}_{t\to 1}), \\ \lambda &= \frac{(1-t)\mathcal{V}_{t\leftarrow 0}}{(1-t)\mathcal{V}_{t\leftarrow 0} + t(1-\mathcal{V}_{t\leftarrow 0})}, \end{aligned} \tag{1}$$

where $\mathcal{F}_{t\to 0}$ and $\mathcal{F}_{t\to 1}$ denote the optical flows from $\mathcal{I}_t$ to $\mathcal{I}_0$ and $\mathcal{I}_t$ to $\mathcal{I}_1$, $g(\cdot,\cdot)$ is an inverse warping function using bilinear interpolation, '$\circ$' denotes the element-wise (Hadamard) product and $\mathcal{V}_{t\leftarrow 0}$ represents the soft visibility map from $\mathcal{C}_0$ to $\mathcal{C}_t$. However, it is difficult to compute the inverse optical flows, *i.e.* $\mathcal{F}_{t\to 0}$ and $\mathcal{F}_{t\to 1}$ in (1), because the target novel view $\mathcal{I}_t$ is unknown. Since the bidirectional optical flows, *i.e.* $\mathcal{F}_{0\to 1}$ and $\mathcal{F}_{1\to 0}$, are much easier to be estimated, the inverse optical flows are typically approximated from them via

$$\begin{aligned} \tilde{\mathcal{F}}_{t\to 0} &= -(1-t)t\mathcal{F}_{0\to 1} + t^2\mathcal{F}_{1\to 0}, \\ \tilde{\mathcal{F}}_{t\to 1} &= (1-t)^2\mathcal{F}_{0\to 1} - t(1-t)\mathcal{F}_{1\to 0}. \end{aligned} \tag{2}$$

3.3. Flow-Assisted Shearlet Transform (FAST)

Inspired by the success of Super-SloMo [8] in video frame interpolation, a novel learning-based method, referred to as Flow-Assisted Shearlet Transform (FAST), is proposed to reconstruct DSLFs from SSLFs. The FAST method adopts a state-of-the-art optical flow approach, *i.e.* PWC-Net [5], to estimate the bidirectional optical flows between neighboring views in a SSLF. Besides, FAST also leverages a state-of-the-art DSLF reconstruction method, *i.e.* ST [4], to guide novel view synthesis. The architecture of FAST is illustrated in Fig. 2. As can be seen from this figure, FAST is composed of two deep convolutional neural networks based on the U-Net architecture [12], which are Disparity Refinement Network (DRN) and View Synthesis Network (VSN). Regarding the architecture of DRN, it has six hierarchies in the encoder part and five hierarchies in the decoder part with the same architecture as the flow interpolation CNN in Super-SloMo. Since the horizontal-parallax SSLF shown in Fig. 1 does not involve any object motion along the vertical axis, only the horizontal components of the bidirectional optical flows estimated by PWC-Net contain useful information, which are denoted by $\tilde{\mathcal{F}}^u_{0\to 1}$ and $\tilde{\mathcal{F}}^u_{1\to 0}$ and called bidirectional disparity maps. The inverse disparity maps, *i.e.* $\tilde{\mathcal{F}}^u_{t\to 0}$ and $\tilde{\mathcal{F}}^u_{t\to 1}$, can then be estimated by using (2). The DRN of FAST takes $\tilde{\mathcal{F}}^u_{0\to 1}$, $\tilde{\mathcal{F}}^u_{1\to 0}$, $\tilde{\mathcal{F}}^u_{t\to 0}$, $\tilde{\mathcal{F}}^u_{t\to 1}$, $g(\mathcal{I}_0, \tilde{\mathcal{F}}^u_{t\to 0})$, $g(\mathcal{I}_1, \tilde{\mathcal{F}}^u_{t\to 1})$, $\mathcal{I}_0$, $\mathcal{I}_1$ and $\mathcal{I}^{\text{ST}}_t$ as the input (19 channels in total) and outputs $\mathcal{F}^u_{t\to 0}$, $\mathcal{F}^u_{t\to 1}$ and $\mathcal{V}_{t\leftarrow 0}$, which are used to interpolate an intermediate view $\hat{\mathcal{I}}_t$ via (1). With regard to the architecture of VSN, it is a "shallow" version of DRN with four hierarchies in the encoder part and three hierarchies in the decoder part. The interpolated novel view $\hat{\mathcal{I}}_t$ and the corresponding view reconstructed by ST, *i.e.* $\mathcal{I}^{\text{ST}}_t$, are fed to the VSN of FAST to generate the final target view $\mathcal{I}_t$.

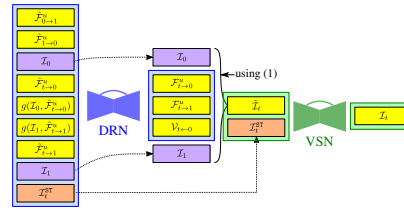


Fig. 2. An overview of the architecture of FAST. Note that $\mathcal{I}^{\text{ST}}_t$ is the view reconstructed by ST and $t \in \left\{\frac{1}{\delta}, \frac{2}{\delta}, ..., \frac{\delta-1}{\delta}\right\}$ (see Sect. 1).

Loss functions. The loss function of FAST is composed of VSN reconstruction loss, DRN reconstruction loss and warping loss, all of which are based on ℓ_1 norm:

$$\mathcal{L}^{\text{FAST}} = \omega_1\mathcal{L}^{\text{VSN}} + \omega_2\mathcal{L}^{\text{DRN}} + \omega_3\mathcal{L}^{\text{W}}, \tag{3}$$

where

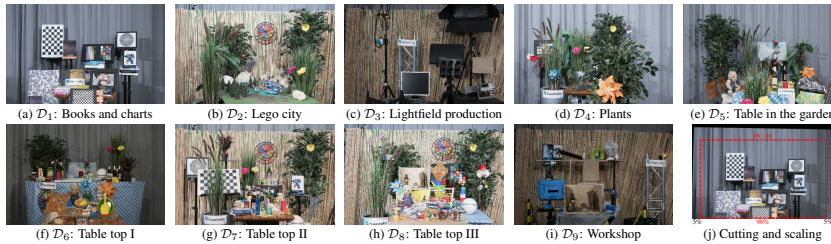
$$\begin{aligned} \mathcal{L}^{\text{VSN}} &= \left\|\mathcal{I}_t - \mathcal{I}^{\text{GT}}_t\right\|_1, \\ \mathcal{L}^{\text{DRN}} &= \left\|\hat{\mathcal{I}}_t - \mathcal{I}^{\text{GT}}_t\right\|_1, \\ \mathcal{L}^{\text{W}} &= \left\|g(\mathcal{I}_0, \mathcal{F}^u_{t\to 0}) - \mathcal{I}^{\text{GT}}_t\right\|_1 + \left\|g(\mathcal{I}_1, \mathcal{F}^u_{t\to 1}) - \mathcal{I}^{\text{GT}}_t\right\|_1, \end{aligned} \tag{4}$$

$\omega_1 = 9$, $\omega_2 = 2$ and $\omega_3 = 1$. Note that these weights are set empirically with a consideration that VSN reconstruction loss is more important than the other two losses.

4. EXPERIMENTS

4.1. Experimental settings

Training dataset. Two light field datasets are used for the training process. One is the Standford light field dataset captured by the Lego gantry with an angular resolution of 17×17. The other is the 4D light field benchmark with an angular resolution of 9×9 created with Blender [13]. The Standford Lego-gantry light field dataset is composed of 13 4D light fields. In each of these light fields, the center region with a size of 512×512 pixels *w.r.t.* each view is cut to make up 17 horizontal-parallax sub-datasets, each of which has 17 parallax images. Note that not all of these 13 4D light fields can be used for the training process, which is because 1) the PWC-Net algorithm fails in disparity estimation for the light field scenes containing reflective and transparent objects; 2) for some scenes, the estimated disparity range is beyond 64 pixels, which will be extremely expensive *w.r.t.* computation time if using ST method; 3) only static scenes are considered here. Therefore, eight 4D light fields are picked out of the Standford light field dataset. For the 4D light field benchmark, it is originally designed for depth estimation from 4D light fields. The



(a) $\mathcal{D}_1$: Books and charts (b) $\mathcal{D}_2$: Lego city (c) $\mathcal{D}_3$: Lightfield production (d) $\mathcal{D}_4$: Plants (e) $\mathcal{D}_5$: Table in the garden

(f) $\mathcal{D}_6$: Table top I (g) $\mathcal{D}_7$: Table top II (h) $\mathcal{D}_8$: Table top III (i) $\mathcal{D}_9$: Workshop (j) Cutting and scaling

Fig. 3. Middle views of evaluation DSLF sub-datasets $\mathcal{D}_\mu$ ($1 \leqslant \mu \leqslant 9$) and (j) illustrates the image cutting and scaling strategy.

benchmark contains four stratified, four test, and four training synthetic light field scenes. Besides, there are 16 additional community-supported photorealistic light field scenes captured in the same way as the benchmark. All of these 28 4D light fields have the same spatial resolution of 512×512 pixels and ground-truth d_{min} and d_{max}, which can facilitate the ST-based DSLF reconstruction. Each 4D light field is then converted into three horizontal-parallax sub-datasets, corresponding to 1st, 5th and 9th rows of views of it.

Training data augmentation. As introduced above, the training dataset consists of 220 horizontal-parallax sub-datasets, of which 136 have the resolution of $17 \times 512 \times 512 \times 3$ and the remaining ones have the resolution of $9 \times 512 \times 512 \times 3$. For each horizontal-parallax sub-dataset, nine continuous parallax images are randomly selected and cropped together into nine image patches with a size of 352×352 pixels during each iteration of the training process. The first and ninth cropped image patches are represented by $\mathcal{I}_0$ and $\mathcal{I}_1$, and used as input views for PWC-Net, ST and FAST. The remaining image patches are denoted by $\mathcal{I}_t$, $t \in \{\frac{1}{8}, \frac{2}{8}, ..., \frac{7}{8}\}$, which are used as the target views to be reconstructed for the neural network training of FAST.

Evaluation dataset. High Density Camera Array (HDCA) dataset is a real-world high-resolution 4D light field dataset captured by a DSLR camera mounted on a precisely-controlled gantry [14]. This dataset is composed of nine different 4D light fields with the same spatial resolution of 3976×2652 pixels. Eight of these light fields have the angular resolution of 101×21 and the remaining one has 99×21 views. Nevertheless, directly using the HDCA dataset is inappropriate for the performance evaluation of different DSLF reconstruction methods, because 1) all 4D light fields in this dataset are not densely-sampled; 2) black borders caused by calibration are not cut out as shown in Fig. 3 (j). Accordingly, a novel cutting and scaling strategy is proposed for transforming all the nine light fields of the HDCA dataset into DSLFs as illustrated in Fig. 3 (j). In particular, a $16:10$ image at the bottom center with occupying 90% of the width of the original view is cut and downscaled to a new resolution of 512×320 pixels by bicubic interpolation. Afterwards, only the top 97 horizontal-parallax views of each light field are used to construct a corresponding 3D DSLF, of which the resolution is $97 \times 512 \times 320 \times 3$. The generated ground-truth evaluation DSLF sub-datasets are denoted by $\mathcal{D}_\mu = \{\mathcal{I}^d_{i,\mu} | 1 \leqslant i \leqslant n\}$, where $\mathcal{I}^d_{i,\mu} \in \mathbb{R}^{M \times N \times 3}$, $1 \leqslant \mu \leqslant 9$, $n = 97$, $M = 512$ and $N = 320$. The middle views of them, *i.e.* $\mathcal{I}^d_{49,\mu}$, are shown in Fig. 3 (a)-(i). After setting the interpolation rate $\delta = 32$, the corresponding evaluation SSLF sub-datasets are represented by $\mathcal{S}_\mu = \{\mathcal{I}^j_{j,\mu} | 1 \leqslant j \leqslant m\}$, where $m = \frac{n-1}{\delta} + 1 (= 4)$. The estimated maximum and minimum disparities and disparity range of each $\mathcal{S}_\mu$ by PWC-Net are illustrated in Fig. 4. It can be found that the disparity-range values

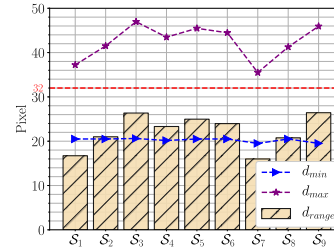


Fig. 4. Disparity estimations of $\mathcal{S}_\mu$ ($1 \leqslant \mu \leqslant 9$) using PWC-Net.

of the nine evaluation SSLF sub-datasets vary from 16 to 26 pixels; consequently, $\frac{d_{range}}{\delta} < 1$ pixel, which suggests that all the ground-truth DSLF sub-datasets $\mathcal{D}_\mu$ ($1 \leqslant \mu \leqslant 9$) are densely sampled.

Evaluation criteria. The per-view PSNR for a ground-truth DSLF sub-dataset $\mathcal{D}_\mu$ and the corresponding DSLF sub-dataset $\tilde{\mathcal{D}}_\mu$ reconstructed from $\mathcal{S}_\mu$ is described as below:

$$\mathrm{MSE}_{i,\mu} = \frac{1}{3 \times M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} \left\| \tilde{\mathcal{I}}^d_{i,\mu}(x,y) - \mathcal{I}^d_{i,\mu}(x,y) \right\|_2^2, \tag{5}$$

$$\mathrm{PSNR}_{i,\mu} = 10 \log_{10} \left(\frac{255^2}{\mathrm{MSE}_{i,\mu}} \right).$$

The minimum per-view PSNR for each $\mathcal{D}_\mu$ is used as the final evaluation criteria as same as [9].

Implementation details. The proposed FAST method is implemented by using PyTorch, where the training mini-batch size is 6, the Adam optimizer is employed for training $1,500$ epochs and the learning rate of it is fixed to be 0.0001. The whole training process takes around 36 hours on an Nvidia Titan Xp GPU. Besides, the parameters of ST using the DORE algorithm are set as same as [4].

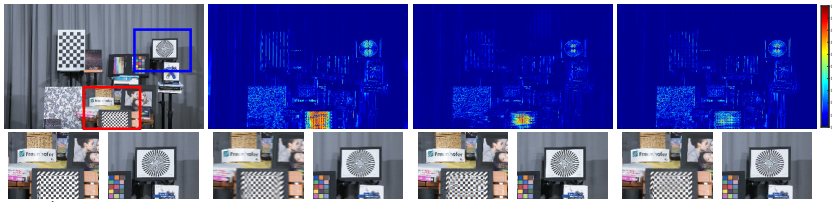
4.2. Results and analysis

The proposed method and other DSLF reconstruction approaches are evaluated quantitatively and qualitatively as follows:

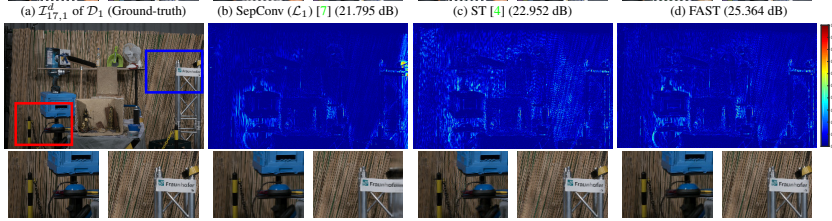
Quantitative evaluation. The minimum per-view PSNR values of the DSLF reconstruction on all the nine evaluation SSLF sub-datasets $\mathcal{S}_\mu$ using different methods are exhibited in Table 1. As can be seen from this table, the proposed FAST method achieves the best performance on most of the evaluation SSLF sub-datasets except for $\mathcal{S}_\mu$, $\mu \in \{4, 5\}$. However, on these two SSLF sub-datasets, the minimum per-view PSNR values of the FAST method are only 0.276 and 0.093 dB less than those of the ST approach. Besides, on

Table I. Minimum per-view PSNR values (in dB, explained in Sect. 4.1) for the performance evaluation of different DSLF reconstruction methods on evaluation SSLF sub-datasets $\mathcal{S}_\mu$ ($1 \leqslant \mu \leqslant 9$).

Method	$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$	$\mathcal{S}_4$	$\mathcal{S}_5$	$\mathcal{S}_6$	$\mathcal{S}_7$	$\mathcal{S}_8$	$\mathcal{S}_9$
SepConv ($\mathcal{L}_1$) [7]	21.018	17.579	20.330	22.215	24.955	25.924	18.141	18.781	22.715
PIASC ($\mathcal{L}_1$) [9]	21.015	17.572	20.332	22.221	24.961	25.929	18.140	18.784	22.709
ST [4]	22.699	17.634	20.231	**23.842**	**25.752**	26.527	18.015	18.727	22.304
FAST	**24.683**	**17.988**	**20.828**	23.566	25.659	**27.002**	**18.393**	**19.171**	**22.884**



(a) $\mathcal{I}^d_{17,1}$ of $\mathcal{D}_1$ (Ground-truth) (b) SepConv ($\mathcal{L}_1$) [7] (21.795 dB) (c) ST [4] (22.952 dB) (d) FAST (25.364 dB)



(e) $\mathcal{I}^d_{29,9}$ of $\mathcal{D}_9$ (Ground-truth) (f) SepConv ($\mathcal{L}_1$) [7] (26.711 dB) (g) ST [4] (26.997 dB) (h) FAST (28.514 dB)

Fig. 5. Visualization of the results of different DSLF reconstruction methods.

$\mathcal{S}_1$, the DSLF reconstruction results indicate that the FAST method achieves a significant improvement in performance over the second-best approach with a gain of 1.984 dB, which demonstrates the effectiveness of the proposed method. Moreover, it can also be seen from the data in this table that SepConv and PIASC behave almost the same and both of them outperform the ST approach on $\mathcal{S}_\mu$, $\mu \in \{3, 7, 8, 9\}$.

Qualitative evaluation. Since the above analysis shows the similar DSLF reconstruction performance results of SepConv and PIASC, only SepConv is visually evaluated here. Some of the reconstructed views from $\mathcal{S}_\mu$ using different methods are displayed in Fig. 5. The top row of this figure illustrates the DSLF reconstruction results *w.r.t.* $\mathcal{I}^d_{17,1}$ of $\mathcal{D}_1$. The image patches containing the checkerboard and Siemens star are selected as the interesting areas. The SepConv method totally fails in reconstructing the checkerboard part with blur and artifacts. In addition, the Siemens star reconstructed by it is also blurry. This is because that the size of the repetitive pattern, *i.e.* the checker, is smaller than the disparities of neighboring views of $\mathcal{S}_\mu$, which confuses the network of SepConv at the aspect of the real-motion decision. The ST approach does not have such motion-decision problem; however, it still outputs artifacts when reconstructing the checkerboard. The proposed FAST method alleviates this problem with visually appealing results for both interesting areas, partially because it exploits the disparity information. The bottom row of Fig. 5 shows the visualization results *w.r.t.* $\mathcal{I}^d_{29,9}$ of $\mathcal{D}_9$. Two image patches involving the toolbox and Fraunhofer logo with a background of bamboo curtain are chosen as the interesting areas. Looking at Fig. 5 (f), the SepConv method succeeds in reconstructing the toolbox part; however, it fails in the reconstruction of the border of the Fraunhofer-logo patch. The ST approach solves this blur-border problem, while it produces artifacts *w.r.t.* the bamboo curtain in the interesting area including the toolbox. The proposed FAST method successfully addresses these issues with achieving sharp and visually correct results for both interesting areas, which implies that FAST is effective in DSLF reconstruction for the scene having a complex background.

5. CONCLUSION

This paper presents a novel learning-based method, Flow-Assisted Shearlet Transform (FAST), for solving the DSLF reconstruction problem. The FAST method fully leverages a state-of-the-art optical flow method, *i.e.* PWC-Net, to estimate bidirectional disparity maps between adjacent views in an input SSLF, which are beneficial to the estimation of the inverse disparity maps and the preparation of the shearlet system in ST. Besides, FAST employs two fully convolutional neural networks, *i.e.* disparity refinement network and view synthesis network, to reconstruct a DSLF from a SSLF using the disparity information and ST results. Experimental results on nine challenging real-world DSLF sub-datasets with large disparity ranges show that the proposed FAST method achieves better DSLF reconstruction results than the other state-of-the-art approaches.

Acknowledgments. The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, and the German Research Foundation (DFG) No. K02044/8-1. The Titan Xp used for this research was donated by the NVIDIA Corporation.

6. REFERENCES

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*, 1996, pp. 31–42. 1

[2] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Image based rendering technique via sparse representation in shearlet domain," in *ICIP*, 2015, pp. 1379–1383. 1

[3] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018. 1, 2

[4] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017. 1, 2, 3, 4

[5] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *CVPR*, 2018, pp. 8934–8943. 1, 2

[6] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *CVPR*, 2017, pp. 2270–2279. 1

[7] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *ICCV*, 2017, pp. 261–270. 1, 4

[8] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super SloMo: High quality estimation of multiple intermediate frames for video interpolation," in *CVPR*, 2018, pp. 9000–9008. 1, 2

[9] Y. Gao and R. Koch, "Parallax view generation for static scenes using parallax-interpolation adaptive separable convolution," in *ICME Workshops*, 2018, pp. 1–4. 1, 3, 4

[10] Y. Gao, R. Bregovic, A. Gotchev, and R. Koch, "MAST: Mask-accelerated shearlet transform for densely-sampled light field reconstruction," in *ICME*, 2019. 1

[11] M. Genzel and G. Kutyniok, "Asymptotic analysis of inpainting via universal shearlet systems," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 2301–2339, 2014. 2

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015, pp. 234–241. 2

[13] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4D light fields," in *ACCV*, 2016, pp. 19–34. 2

[14] M. Ziegler, R. op het Veld, J. Keinert, and F. Zilly, "Acquisition system for dense lightfield of large scenes," in *3DTV-CON*, 2017, pp. 1–4. 3

6.10 Publication 10

DRST: Deep Residual Shearlet Transform for Densely-Sampled Light Field Reconstruction

Yuan Gao, Robert Bregovic, Reinhard Koch and Atanas Gotchev

Available in

arXiv preprint arXiv:2003.08865

DRST: Deep Residual Shearlet Transform for Densely-Sampled Light Field Reconstruction

Yuan Gao, Robert Bregović, *Member, IEEE,* Reinhard Koch, *Member, IEEE,* and Atanas Gotchev, *Member, IEEE*

***Abstract*—The Image-Based Rendering (IBR) approach using Shearlet Transform (ST) is one of the most effective methods for Densely-Sampled Light Field (DSLF) reconstruction. The ST-based DSLF reconstruction typically relies on an iterative thresholding algorithm for Epipolar-Plane Image (EPI) sparse regularization in shearlet domain, involving dozens of transformations between image domain and shearlet domain, which are in general time-consuming. To overcome this limitation, a novel learning-based ST approach, referred to as Deep Residual Shearlet Transform (DRST), is proposed in this paper. Specifically, for an input sparsely-sampled EPI, DRST employs a deep fully Convolutional Neural Network (CNN) to predict the residuals of the shearlet coefficients in shearlet domain in order to reconstruct a densely-sampled EPI in image domain. The DRST network is trained on synthetic Sparsely-Sampled Light Field (SSLF) data only by leveraging elaborately-designed masks. Experimental results on three challenging real-world light field evaluation datasets with varying moderate disparity ranges (8 - 16 pixels) demonstrate the superiority of the proposed learning-based DRST approach over the non-learning-based ST method for DSLF reconstruction. Moreover, DRST provides a 2.4x speedup over ST, at least.**

***Index Terms*—Densely-sampled light field reconstruction, novel view synthesis, epipolar-plane image, Shearlet Transform (ST), Deep Residual Shearlet Transform (DRST).**

I. Introduction

DENSELY-SAMPLED Light Field (DSLF) is a discrete representation of the 4D approximation of the plenoptic function parameterized by two parallel planes (camera plane and image plane) [1], where multi-perspective camera views are arranged in such a way that the disparity ranges between adjacent views are less than or equal to one pixel [2]. DSLF has a wide range of applications, such as depth estimation, super-resolution and synthetic aperture imaging [3], visualization on 3DTV [4] and Virtual Reality (VR) [5] devices. In real-world environments, a DSLF is extremely difficult to capture by modern light field acquisition systems, such as micro-lens array (MLA) [6, 7], multi-camera array [8]–[10] and coded mask [11, 12]. Nevertheless, these state-of-the-art light field devices are successful in capturing Sparsely-Sampled Light Fields (SSLFs), where the disparity ranges of any two neighboring views are larger than one pixel. Therefore, for real-world scenes, DSLFs are typically reconstructed from SSLFs. This paper studies how to effectively and efficiently reconstruct a DSLF for a real-world SSLF.

The Shearlet Transform (ST)-based DSLF reconstruction algorithm [13, 14] is one of the state-of-the-art Image-Based Rendering (IBR) approaches [15, 16], which treats an input SSLF as a set of sparsely-sampled Epipolar-Plane Images (EPIs) and leverages the sparse representation of these EPIs in shearlet domain to perform densely-sampled EPI reconstruction in image domain. However, the sparse regularization by ST is an iterative algorithm that involves dozens of iterations of domain transformations, *i.e.* shearlet analysis transform from image domain to shearlet domain and shearlet synthesis transform from shearlet domain to image domain. To be more precise, a shearlet analysis transform converts an input grayscale EPI into η shearlet coefficients, which requires one 2D Discrete Fourier Transform (DFT) and η 2D inverse DFTs. On the contrary, a shearlet synthesis transform converts the regularized η shearlet coefficients into the output grayscale EPI, requiring η 2D DFTs and one 2D inverse DFT. As a result, ST tends to be time-consuming for DSLF reconstruction on SSLFs with large spatial or large angular resolution.

To address this fundamental issue, a novel learning-based approach, referred to as Deep Residual Shearlet Transform (DRST), is proposed in this paper. In particular, DRST performs shearlet coefficient reconstruction in shearlet domain for an input sparsely-sampled EPI by means of a deep Convolutional Neural Network (CNN), which is composed of a residual learning strategy and an encoder-decoder network that predicts the residuals of the shearlet coefficients. The reconstructed shearlet coefficients in shearlet domain are then transformed back into image domain to produce a corresponding inpainted densely-sampled EPI. Finally, a target DSLF can be reconstructed by repeating this EPI reconstruction process on all the sparsely-sampled EPIs of the input SSLF. Besides, the network of DRST is fully convolutional and end-to-end trainable. Considering the aforementioned difficulty of acquiring ground-truth DSLFs, the training of DRST is performed on SSLF data only. The synthetic SSLF data are used for training because the ground-truth disparity information, which is important to the shearlet system construction, pre- and post-shearing steps of DRST, can be provided by using the state-of-the-art 3D computer graphics softwares.

The key contributions of this paper are as follows.

- We propose a learning-based DRST method that achieves better DSLF reconstruction performance than the non-learning-based ST algorithm on three evaluation datasets

Y. Gao, R. Bregović and A. Gotchev are with the Faculty of Information Technology and Communication Sciences (ITC), Tampere University, 33014 Tampere, Finland. (e-mail: {yuan.gao, robert.bregovic, atanas.gotchev}@tuni.fi)

R. Koch is with the Department of Computer Science, Kiel University, 24118 Kiel, Germany. (e-mail: rk@informatik.uni-kiel.de)

The work in this paper was funded from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 676401, European Training Network on Full Parallax Imaging, and the German Research Foundation (DFG) No. K02044/8-1. The authors thank Nvidia for GPU hardware donations.

composed of real-world horizontal-parallax light fields with different moderate disparity ranges (8 - 16 pixels);
- The network of DRST is trained on synthetic SSLF data by means of the elaborately-designed masks. To our best knowledge, this is the first work to investigate learning-based DSLF reconstruction with only exploiting synthetic SSLFs as training data;
- The proposed learning-based DRST is more time-efficient than the non-learning-based ST. Specifically, DRST provides a 2.4x speedup over ST, at least.

The paper is organized as follows. Section II first introduces the related work on DSLF reconstruction and then outlines how to employ the non-learning-based ST for DSLF reconstruction. In Section III, we detail the proposed learning-based DRST. Section IV is devoted to the experiments and analysis of DRST and other baseline approaches. Finally, Section V concludes and summarizes this paper.

II. Related Work

As pointed out in the introduction to this paper, the modern light field acquisition systems can hardly capture DSLFs in real-world environments due to their hardware limitations; however, a real-world SSLF with a moderate disparity range (8 - 16 pixels) is possible to capture by most of them. Therefore, performing an effective and efficient DSLF reconstruction on the captured SSLFs with moderate disparity ranges is the best way to compensate for the hardware limitations of these modern light field acquisition systems. The DSLF reconstruction problem can potentially be solved by several approaches that are categorized into two types, *i.e.* learning-based novel view synthesis and light field angular super-resolution. Regarding the former type, Niklaus *et al.* propose a spatially-adaptive Separable Convolution (SepConv) approach that employs a CNN to predict the separable 1D kernels for video frame synthesis [17]. Gao and Koch propose a fine-tuning strategy for SepConv, referred to as Parallax-Interpolation Adaptive Separable Convolution (PIASC), to generate novel parallax views for the input SSLF in a recursive manner [18]. With regard to the latter type, Kalantari *et al.* propose a learning-based view synthesis method, consisting of disparity and color estimation components, to synthesize novel views for a MLA-based consumer light field camera [19]. Wu *et al.* leverage a CNN with a residual learning strategy to perform angular detail restoration on EPIs; however, the maximum disparity range of the input SSLF that can be handled by this method is only 5 pixels [20]. More recently, Yeung *et al.* also exploit an end-to-end CNN, consisting of the view synthesis and refinement networks, for light field angular resolution enhancement in a coarse-to-fine manner [21]. Nevertheless, this method cannot be directly used to solve the DSLF reconstruction problem because their networks rely on a fixed interpolation rate δ (see Section IV-A), while this rate is generally much smaller than the sampling interval τ (introduced in the next section) for a target DSLF to be reconstructed. Wang *et al.* propose a 4D CNN to enhance the angular resolution of an input 4D SSLF [22]; however, the interpolation rate δ of this approach is either 2 or 3 ($\ll \tau$). The

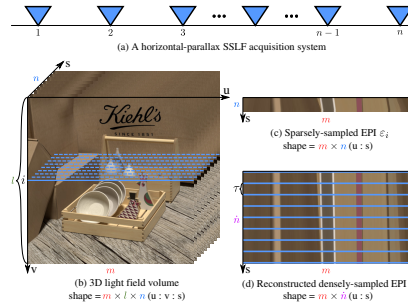


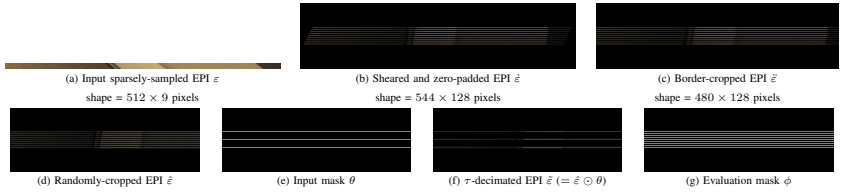
Figure 1. Introduction to the DSLF reconstruction problem.

ST-based IBR algorithm [13, 14] is the first method especially designed for solving the DSLF reconstruction problem. In particular, ST fully leverages the light field sparsification in shearlet domain to perform image inpainting on the sparsely-sampled EPIs of the input SSLF [23]. Since the proposed learning-based DRST is partially based on the non-learning-based ST, a brief introduction to ST is presented as follows.

Shearlet Transform (ST) [24, 25] is adapted to perform DSLF reconstruction on SSLFs by leveraging the sparsity of EPIs in shearlet domain [13, 14]. Typically, the ST-based DSLF reconstruction comprises four steps: (i) pre-shearing, (ii) shearlet system construction, (iii) sparse regularization and (iv) post-shearing. Steps (i), (ii) and (iv) require the disparity estimation of the input SSLF, *i.e.* the minimal disparity d_{min}, maximal disparity d_{max} and disparity range $d_{range} = (d_{max} - d_{min})$. The estimated disparity data are employed to rearrange the rows of each sparsely-sampled EPI via shearing and zero padding operations and to construct a specifically-tailored universal shearlet system with ξ scales, where $\xi = \lceil \log_2 \tau \rceil$. The sparse regularization step is the core of ST, consisting of (i) shearlet analysis transform, (ii) hard thresholding, (iii) shearlet synthesis transform and (iv) double overrelaxation (DORE) [14]. To be more precise, shearlet analysis transform transforms an EPI in image domain into shearlet coefficients in shearlet domain, hard thresholding performs regularization on the transformed coefficients in shearlet domain, shearlet synthesis transform transforms the regularized coefficients into a processed EPI in image domain, and DORE is an optional algorithm accelerating the convergence speed of the whole sparse regularization step. Moreover, the sparse regularization step is an iterative algorithm, *i.e.* for each color channel of each pre-sheared and zero-padded sparsely-sampled input EPI, this step is repeated typically 50 - 100 times, thereby affecting the time efficiency of ST when reconstructing DSLFs from SSLFs of challenging light field scenes that require a high number of iterations.

III. Deep Residual Shearlet Transform (DRST)

Inspired by the above ST-based DSLF reconstruction, a novel learning-based ST approach, referred to as DRST, is proposed by fully leveraging the state-of-the-art deep learning



(a) Input sparsely-sampled EPI ε shape = 512×9 pixels

(b) Sheared and zero-padded EPI $\dot{\varepsilon}$ shape = 544×128 pixels

(c) Border-cropped EPI $\ddot{\varepsilon}$ shape = 480×128 pixels

(d) Randomly-cropped EPI $\hat{\varepsilon}$ shape = 384×128 pixels

(e) Input mask θ shape = 384×128 pixels

(f) τ-decimated EPI $\bar{\varepsilon}$ $(= \hat{\varepsilon} \odot \theta)$ shape = 384×128 pixels

(g) Evaluation mask ϕ shape = 384×128 pixels

Figure 2. Training data preparation. A sparsely-sampled EPI ε from a training 3D SSLF is illustrated in (a). The sheared and zero-padded EPI $\dot{\varepsilon}$ in (b) is the result of performing the pre-shearing and zero-padding step on ε. The border of $\dot{\varepsilon}$ is cut to generate a border-cropped EPI $\ddot{\varepsilon}$ in (c). A random cropping operation is then performed on $\ddot{\varepsilon}$ to produce a $3:1$ randomly-cropped EPI $\hat{\varepsilon}$ presented in (d). For $\hat{\varepsilon}$, an input mask θ is designed as shown in (e) and utilized to generate the τ-decimated EPI $\bar{\varepsilon}$ in (f), which is the input data for the learning-based sparse regularization step of DRST. Finally, the evaluation mask ϕ in (g) is employed to calculate the loss function (4) of the learning-based sparse regularization step.

techniques. Specifically, DRST also consists of four steps: (i) shearlet system construction, (ii) pre-shearing and zero-padding, (iii) learning-based sparse regularization and (iv) post-shearing. The details of these four steps will be elaborated after defining the light field-associated symbols and notations.

A. Symbols and notations

As illustrated in Fig. 1 (a), an input horizontal-parallax SSLF is essentially a set of images uniformly sampled along the horizontal axis s. After stacking all the images of the input 3D SSLF along axis s, a 3D light field volume can be generated as shown in Fig. 1 (b). The generated 3D light field volume has a spatial resolution of $m \times l$ pixels and an angular resolution of n pixels. Let the input 3D SSLF be denoted by $\mathcal{S} = \{\varepsilon_i | 1 \leqslant i \leqslant l\}$, where $\varepsilon_i \in \mathbb{R}^{m \times n \times 3}$ represents a sparsely-sampled EPI. To better understand the EPI structure, one of the sparsely-sampled EPIs of $\mathcal{S}$, *i.e.* ε_i, is picked up from the 3D light field volume in Fig. 1 (b) and shown in Fig. 1 (c). Similarly, the target DSLF to be reconstructed from $\mathcal{S}$ is represented by $\mathcal{D} = \{\zeta_i | 1 \leqslant i \leqslant l\}$, where $\zeta_i \in \mathbb{R}^{m \times \dot{n} \times 3}$ stands for a densely-sampled EPI. It should be noted that each densely-sampled EPI in $\mathcal{D}$ is reconstructed from a corresponding sparsely-sampled EPI in $\mathcal{S}$. The densely-sampled EPI ζ_i, corresponding to ε_i, is presented in Fig. 1 (d). Comparing these two EPIs, it can be found that ζ_i has a higher resolution than ε_i along the s axis. Specifically, the number of rows of ζ_i, *i.e.* $\dot{n}$, is decided by the sampling interval τ and the number of rows of ε_i, *i.e.* n, with an equation $\dot{n} = \big((n-1)\tau + 1\big)$. In other words, for the same input SSLF $\mathcal{S}$, the angular resolution of the target DSLF $\mathcal{D}$ to be reconstructed depends on the sampling interval τ that is controlled by the disparity range (d_{range}) of $\mathcal{S}$, *i.e.* $\tau \geqslant d_{range}$. In this paper, we target solving the DSLF reconstruction problem for any input SSLF $\mathcal{S}$ with a moderate disparity range, *i.e.* $8 < d_{range} \leqslant 16$ pixels. In addition, as mentioned in the previous section, the number of the scales of the target shearlet system, ξ, relies on the sampling interval τ. In particular, $\xi = 4$ when $8 < \tau \leqslant 16$, and $\xi = 5$ when $16 < \tau \leqslant 32$. For the shearlet system construction and learning-based sparse regularization steps of DRST, using a shearlet system with 4 scales is much faster than using a shearlet system with 5 scales. As a result, the sampling interval τ is set to 16 for this paper.

B. Shearlet system construction

The specifically-tailored universal shearlet system in [13] is chosen to be constructed for the shearlet analysis and synthesis transforms in the learning-based sparse regularization step. A shearlet analysis transform is denoted by $\mathcal{SH} : \mathbb{R}^{\gamma \times \gamma} \rightarrow \mathbb{R}^{\gamma \times \gamma \times \eta}$, where $\gamma \times \gamma$ represents the size of a shearlet filter and η denotes the number of shearlets in a shearlet system. A shearlet synthesis transform is represented by $\mathcal{SH}^* : \mathbb{R}^{\gamma \times \gamma \times \eta} \rightarrow \mathbb{R}^{\gamma \times \gamma}$. Note that the number of shearlets, *i.e.* η, is decided by the number of the scales, *i.e.* ξ, of the target shearlet system with an equation $\eta = (2^{\xi+1} + \xi - 1)$. In addition, as described in the previous section, ξ is decided by the sampling interval τ. In our case, $\xi = \lceil \log_2 \tau \rceil = 4$ and, consequently, the target shearlet system has $\eta = 35$ shearlets. The size of the shearlet filters in the target shearlet system is specified by the users, *i.e.* $\gamma = 127$ for this paper.

C. Pre-shearing and zero-padding

For better understanding the pre-shearing and zero-padding strategies and how to leverage the synthetic SSLF data for training, in this section the first-row horizontal-parallax light field of the 4D light field "Boxes" [26] is selected as the input 3D SSLF $\mathcal{S}$ for demonstration. The input 3D light field $\mathcal{S}$ has an angular resolution 9 pixels and a spatial resolution 512×512 pixels. The ground-truth disparity information of $\mathcal{S}$ is provided by the dataset, *i.e.* $d_{min} = -2.2$, $d_{max} = 1.4$ and $d_{range} = 3.6$ pixels. The first sparsely-sampled EPI of $\mathcal{S}$, represented by ε, is illustrated in Fig. 2 (a). It can be seen that the shape of ε is 512×9 pixels. The values of d_{min} and d_{range} are utilized to shear and pad ε as shown in Fig. 2 (b). Specifically, the sheared and zero-padded EPI $\dot{\varepsilon}$ has nine separated non-black lines from ε. The horizontal and vertical displacements between neighboring non-black lines are φ and $\frac{\tau}{4} = 4$ pixels, respectively. Here, $\big(d_{min} - (\frac{\tau}{4} - d_{range})\big) \leqslant \varphi \leqslant d_{min}$ and $d_{range} \leqslant \frac{\tau}{4}$, are such that the image inpainting on $\dot{\varepsilon}$ can produce a densely-sampled EPI. Moreover, the size of $\dot{\varepsilon}$ is 544×128 pixels. The left and right borders of $\dot{\varepsilon}$ are then cut to generate a border-cropped EPI $\ddot{\varepsilon}$ shown in Fig. 2 (c) with a shape 480×128 pixels. In order to augment training data, a 384×128 - pixels EPI $\hat{\varepsilon}$ is randomly cropped from $\ddot{\varepsilon}$ for each training iteration. Note that $\ddot{\varepsilon}$ and $\hat{\varepsilon}$ have the same height, implying that the random cropping operation here is essentially

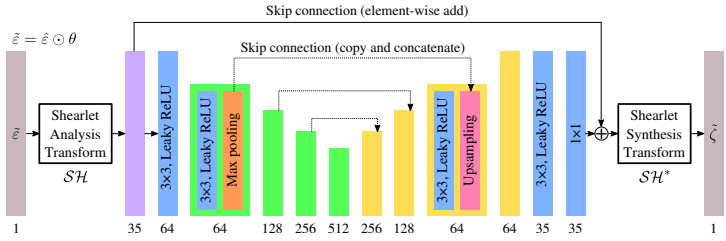


Figure 3. Network architecture of the learning-based sparse regularization in DRST.

to slide a 3 : 1 window inside $\tilde{\varepsilon}$ along the horizontal axis to produce one crop, *i.e.* $\hat{\varepsilon}$, which is illustrated in Fig. 2 (d).

Masks. There are two masks, input mask θ and evaluation mask ϕ, associated with the cropped EPI $\hat{\varepsilon}$ and prepared for the learning-based sparse regularization step. In particular, the input mask θ has three non-zero lines, corresponding to the first, middle and last non-zero lines of $\hat{\varepsilon}$. The evaluation mask ϕ has nine non-zero lines, corresponding to all the nine non-zero lines of $\hat{\varepsilon}$. The input mask θ and evaluation mask ϕ are illustrated in Fig. 2 (e) and (g), respectively. In addition, the input mask θ is utilized to generate a sparsely-sampled EPI $\tilde{\varepsilon}$ from the cropped EPI $\hat{\varepsilon}$ as the input data for the sparse regularization of ST and learning-based sparse regularization of DRST, *i.e.* $\tilde{\varepsilon} = \hat{\varepsilon} \odot \theta$, where $\odot$ denotes the Hadamard product. The vertical displacement between any two adjacent non-zero lines of the input mask θ or sparsely-sampled $\tilde{\varepsilon}$ is equal to the sampling interval τ. As a result, $\tilde{\varepsilon}$ is also referred to as τ-decimated EPI, which is illustrated in Fig. 2 (f).

D. Learning-based sparse regularization

The goal of the sparse regularization step in ST is to reconstruct a densely-sampled EPI $\tilde{\zeta}$ from the above generated τ-decimated EPI $\tilde{\varepsilon}$. This can be achieved by solving the following optimization problem in the shearlet transform domain:

$$\min_{\tilde{\zeta}} \left\| \mathcal{SH}(\tilde{\zeta}) \right\|_1, \text{ s.t. } \tilde{\varepsilon} = \theta \odot \tilde{\zeta}. \tag{1}$$

The sparse regularization is an iterative algorithm that solves the above problem by performing regularization on the transform domain coefficients. Different from the sparse regularization step of ST, the learning-based sparse regularization step of DRST is a more efficient non-iterative algorithm, which is introduced as below:

Network architecture. The learning-based sparse regularization in DRST is a deep CNN consisting of an encoder-decoder network and a residual learning strategy, which are inspired by U-Net [27] and ResNet [28], respectively. The network architecture of this CNN is presented in Fig. 3. As shown in this figure, the input data is the τ-decimated EPI $\tilde{\varepsilon}$ and the output data is the reconstructed densely-sampled EPI $\tilde{\zeta}$. The shearlet analysis transform converts $\tilde{\varepsilon}$ into 35 - channels shearlet coefficients in shearlet domain. These coefficients are then fed to the encoder-decoder network to predict residual shearlet coefficients. Specifically, the encoder-decoder network is a U-Net with an encoder having 4 hierarchies and a decoder also having 4 hierarchies. The encoder and decoder in the U-Net are connected by three skip connections (copy and concatenate) at the same spatial resolution for the first three hierarchies. Each hierarchy in the encoder is composed of three layers, *i.e.* a 2D convolution layer, a Leaky ReLU layer ($\alpha = 0.3$) and a max pooling layer for decreasing the spatial resolution by 2. Each hierarchy in the decoder also consists of three layers, *i.e.* a 2D convolution layer, a Leaky ReLU layer ($\alpha = 0.3$) and an upsampling layer with nearest interpolation for increasing the spatial resolution by 2. The convolution kernel size is set to 3×3 for all the 2D convolution layers except for the last one, where the convolution kernel size is set to 1×1. In addition, no Leaky ReLU layer is added behind the last 2D convolution layer. Afterwards, the residual learning strategy is utilized to add the predicted residual shearlet coefficients back to the original shealet coefficients by means of the other type of skip connection, *i.e.* an element-wise add operation. Finally, these processed shearlet coefficients are transformed back to image domain to generate $\tilde{\zeta}$ via the shearlet synthesis transform. Mathematically, the learning-based sparse regularization can be written as below:

$$\tilde{\zeta} = \mathcal{SH}^* \Big(\mathcal{SH}(\tilde{\varepsilon}) + \mathcal{R}\big(\mathcal{SH}(\tilde{\varepsilon})\big) \Big), \tag{2}$$

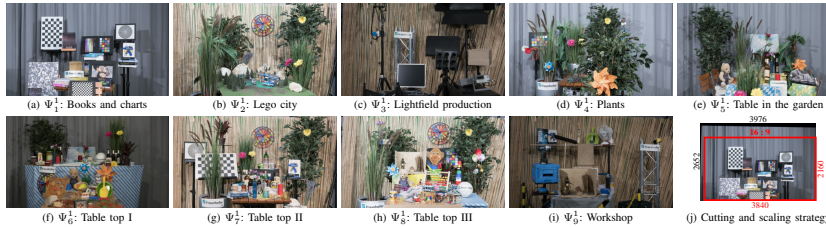
where $\mathcal{R}$ denotes the encoder-decoder network, *i.e.* $\mathcal{R} : \mathbb{R}^{128\times384\times35} \to \mathbb{R}^{128\times384\times35}$ for the training case.

Loss function. The trainable parameters in $\mathcal{R}$ are learned by solving the following optimization problem:

$$\min_{\mathcal{R}} \left\| \tilde{\zeta} - \tilde{\zeta}^{\mathrm{GT}} \right\|_1. \tag{3}$$

However, the ground-truth densely-sampled EPI $\tilde{\zeta}^{\mathrm{GT}}$ corresponding to the reconstructed densely-sampled EPI $\tilde{\zeta}$ is unknown, since the training synthetic SSLF dataset does not offer the corresponding ground-truth DSLF data. Besides, rendering a high-quality and high-resolution synthetic DSLF dataset is prohibitively expensive compared to the rendering of a synthetic SSLF dataset. Therefore, a new loss function without relying on the DSLF data is proposed. Specifically, the loss function for the training of the encoder-decoder network in the learning-based sparse regularization takes account of minimizing the reconstruction error between the ground-truth sparsely-sampled EPI $\tilde{\varepsilon}$ and the reconstructed densely-sampled EPI $\tilde{\zeta}$ using the evaluation mask ϕ via ℓ_1 norm, *i.e.*

$$\mathcal{L} = \left\| \tilde{\varepsilon} - \tilde{\zeta} \odot \phi \right\|_1. \tag{4}$$



(a) Ψ_1^1: Books and charts (b) Ψ_2^1: Lego city (c) Ψ_3^1: Lightfield production (d) Ψ_4^1: Plants (e) Ψ_5^1: Table in the garden

(f) Ψ_6^1: Table top I (g) Ψ_7^1: Table top II (h) Ψ_8^1: Table top III (i) Ψ_9^1: Workshop (j) Cutting and scaling strategy

Figure 4. Middle views of ground-truth light fields Ψ_e^1 ($1 \leqslant e \leqslant 9$) in the evaluation dataset 1. Sub-image (j) illustrates the image cutting and scaling strategy in Section IV-A2.

Table I
DISPARITY ESTIMATION, MINIMUM AND AVERAGE PER-VIEW PSNR RESULTS (IN DB, EXPLAINED IN SECTION IV-B1) FOR THE PERFORMANCE EVALUATION OF DIFFERENT LIGHT FIELD RECONSTRUCTION METHODS ON THE EVALUATION DATASET 1.

. e of $\mathcal{S}_e^1$	Disparity Estimation (pixel)			Minimum PSNR / Average PSNR (dB)			
	$d^{\mathcal{S}}_{min}$	$d^{\mathcal{S}}_{max}$	$d^{\mathcal{S}}_{range}$	SepConv ($\mathcal{L}_1$) [17]	PIASC ($\mathcal{L}_1$) [18]	ST [14]	DRST
1	12.5	22	9.5	21.963 / 24.907	21.957 / 24.905	35.277 / 38.611	**38.241 / 39.933**
2	13.5	24.5	11	26.562 / 29.073	26.579 / 29.117	27.376 / **29.831**	**27.698** / 29.820
3	14	27.5	13.5	30.508 / 32.874	30.528 / 32.952	**32.092** / **34.221**	31.568 / 33.133
4	12.5	27.5	15	31.536 / 34.804	31.584 / 34.986	32.603 / **36.258**	**33.519** / 36.220
5	12.5	27	14.5	32.278 / 33.803	32.327 / 33.926	32.812 / **35.372**	**34.020** / 35.239
6	12.5	27	14.5	30.100 / 32.539	30.108 / 32.605	36.412 / 40.423	**40.237 / 41.505**
7	13	21.5	8.5	26.609 / 29.939	26.621 / 29.982	28.126 / 30.367	**28.973 / 30.759**
8	14	24.5	10.5	26.885 / 29.480	26.910 / 29.536	28.165 / 30.312	**29.529 / 31.061**
9	14	27.5	13.5	33.043 / 35.533	33.078 / 35.672	34.242 / **36.619**	**35.004** / 36.177

Although the ground-truth sparsely-sampled EPI $\hat{\varepsilon}$ is not densely-sampled, it contains 6 non-zero lines that the input τ-decimated EPI $\tilde{\varepsilon}$ does not have, thereby guiding the optimization process for the training of the network of the learning-based sparse regularization.

E. Post-shearing

The target DSLF can be reconstructed after compensating for the horizontal displacement produced by the aforementioned pre-shearing strategy, *i.e.* φ described in Section III-C, for all the reconstructed densely-sampled EPIs. More details can also be found in [29].

IV. EXPERIMENTS

A. Experimental Settings

As explained in the introduction section, the proposed DRST approach is trained on a synthetic SSLF dataset with ground-truth disparity information and evaluated on three challenging real-world light field evaluation datasets. Since both ST and DRST are designed for reconstructing light fields that are densely-sampled, there are two requirements, *i.e.* (a) $d_{range} \leqslant \frac{\tau}{4}$ and (b) $d^{\mathcal{S}}_{range} \leqslant \tau$, for the training and evaluation datasets, respectively. Besides, the interpolation rate δ denotes the sampling rate for extracting a SSLF $\mathcal{S}$ from a ground-truth light field Ψ in an evaluation dataset [30]. More details about the preparation of the training and evaluation datasets and the implementation of DRST are presented next.

1) Training dataset: The 4D light field dataset [26] is a synthetic dataset created with Blender. It is composed of 28 4D light fields of the same size, *i.e.* $9 \times 9 \times 512 \times 512 \times 3$. Among them, there are 18 4D light fields suitable for the network training of DRST, since (i) the four light fields in the category "Stratified" differ a lot from real-world light fields; (ii) the 4D light field "Museum" is rendered for a non-Lambertian scene, where the shadows on the glass lead to a real d_{min} that is lower than the ground-truth d_{min} provided by the dataset; (iii) the 4D light fields "Herbs", "Antinous", "Dishes", "Greek" and "Tower" do not satisfy the requirement (a). The 18 suitable 4D light fields are split into both horizontal- and vertical-parallax SSLFs for a total of $18 \times (9 + 9) = 324$ sets. Note that all the vertical-parallax SSLFs are turned into horizontal-parallax SSLFs by performing $90°$ anticlockwise rotation on all the parallax images. The generated 3D SSLFs $\mathcal{S}_t$ ($1 \leqslant t \leqslant 324$) have the same angular and spatial resolutions, *i.e.* $n = 9$, $l = 512$ and $m = 512$. To augment the number of training samples, the pre-shearing strategy in Section III-C is repeated three times for each $\mathcal{S}_t$, corresponding to the horizontal displacements $\varphi = d_{min}$, $d_{min} - 0.5 \cdot (\frac{\tau}{4} - d_{range})$ and $d_{min} - (\frac{\tau}{4} - d_{range})$, respectively. As a result, 972 sheared input SSLFs are generated, producing $972 \times 512 = 497,664$ border-cropped EPIs for training, of which an example is displayed in Fig. 2 (c) .

2) Evaluation Dataset 1: The High Density Camera Array (HDCA) dataset [31] is a real-world 4D light field dataset captured by a DSLR camera mounted on a high-precision gantry. This dataset is composed of eight light fields of size $101 \times 21 \times 3976 \times 2652 \times 3$ and one light field with a size $99 \times 21 \times 3976 \times 2652 \times 3$. Note that these raw light field data can hardly be used for evaluation for two reasons: (i) parallax images in these light fields have black borders due to calibration (see Fig. 4 (j)); (ii) d_{range} between neighboring views is up to around 5 pixels, suggesting that these light fields are very SSLFs that can not provide enough ground-

(a) Ψ_1^2: Toys (b) Ψ_2^2: Bikes (c) Ψ_3^2: Mannequin (d) Ψ_4^2: Livingroom (e) Ψ_5^2: Workshop

Figure 5. Middle views of ground-truth light fields Ψ_e^2 ($1 \leqslant e \leqslant 5$) in the evaluation dataset 2.

Table II
DISPARITY ESTIMATION, INTERPOLATION RATE, MINIMUM AND AVERAGE PER-VIEW PSNR RESULTS (IN DB, EXPLAINED IN SECTION IV-B1) FOR THE PERFORMANCE EVALUATION OF DIFFERENT LIGHT FIELD RECONSTRUCTION METHODS ON THE EVALUATION DATASET 2.

e of Ψ_e^2	Disparity Estimation (pixel)			Interpolation rate	Minimum PSNR / Average PSNR (dB)			
	d_{min}^{Ψ}	d_{max}^{Ψ}	d_{range}^{Ψ}	δ^2	SepConv ($\mathcal{L}_1$) [17]	PIASC ($\mathcal{L}_1$) [18]	ST [14]	DRST
1	-1.03125	1.0625	2.09375	4	36.427 / 41.554	**36.500** / **41.769**	36.215 / 40.233	36.492 / 41.045
2	-0.875	0.59375	1.46875	8	33.765 / 36.597	33.884 / **36.665**	32.876 / 36.092	**33.971** / 36.435
3	-0.46875	0.4375	0.90625	16	35.757 / 37.658	35.795 / **37.915**	34.426 / 37.433	**35.814** / 37.847
4	-0.375	0.5	0.875	16	40.507 / 42.867	**40.636** / **43.631**	39.064 / 42.028	40.167 / 43.079
5	-0.40625	1.03125	1.4375	8	36.901 / 40.277	37.026 / **40.576**	35.590 / 39.469	**39.274** / 40.354

truth data for the performance evaluation of the DSLF reconstruction approaches. Therefore, a cutting and scaling strategy is proposed to tailor this dataset for the evaluation purpose. Specifically, the top 97 horizontal-parallax images of each light field are processed by the cutting operation represented by the 16 : 9 red box in Fig. 4 (j) and then downscaled to a new resolution, *i.e.* 1280×720 pixels, using the cubic spline kernel [32]. Consequently, the evaluation dataset 1 is composed of nine horizontal-parallax ground-truth light fields Ψ_e^1 ($1 \leqslant e \leqslant 9$) with the same angular and spatial resolutions, *i.e.* $\dot{n}^1 = 97$, $l^1 = 720$ and $m^1 = 1280$. The middle views of these nine ground-truth light fields are shown in Fig. 4 (a) - (i). The interpolation rate δ^1 for uniformly sampling an input SSLF $\mathcal{S}_e^1$ from a ground-truth light field Ψ_e^1 is set to 8, such that nine input SSLFs $\mathcal{S}_e^1$ ($1 \leqslant e \leqslant 9$) are generated with the same angular resolution $n^1 = \left(1 + \frac{\ddot{n}^1 - 1}{\delta^1}\right) = 13$. The disparity information of each $\mathcal{S}_e^1$ is first estimated automatically using a state-of-the-art optical flow method, PWC-Net [33], and then refined manually. The final approximated $d_{min}^{\mathcal{S}}$, $d_{max}^{\mathcal{S}}$ and $d_{range}^{\mathcal{S}}$ for all the input SSLFs are exhibited in the left part of Table I, where $d_{range}^{\mathcal{S}}$ varies from 8.5 to 15 pixels, satisfying the aforementioned requirement (b). The target DSLF $\mathcal{D}_e^1$ to be reconstructed from an input SSLF $\mathcal{S}_e^1$ consists of $\dot{n}^1 = \big((n^1 - 1)\tau + 1\big) = 193$ horizontal-parallax images.

3) Evaluation Dataset 2: The MPI light field archive contains five real-world horizontal-parallax light fields captured by one-meter long motorized linear stage [34]. Each source 3D light field is composed of 101 horizontal-parallax images of the same resolution, *i.e.* 960×720 pixels. Following the same dataset preparation process as above, the top 97 images are chosen to form a ground-truth light field from each source 3D light field. Therefore, the evaluation dataset 2 has five horizontal-parallax ground-truth light fields Ψ_e^2 ($1 \leqslant e \leqslant 5$) with the same angular and spatial resolutions, *i.e.* $\ddot{n}^2 = 97$, $l^2 = 720$ and $m^2 = 960$. The middle views of these five horizontal-parallax ground-truth light fields are exhibited in Fig. 5. Regarding the disparity estimation of these five ground-truth 3D light fields, the interpolation rate δ is first set to 32 to generate five SSLFs with $n = 7$ parallax images having large disparities. The d_{min} and d_{max} of these five generated SSLFs are then estimated by hands with one-pixel measurement resolution. Finally, these estimated d_{min} and d_{max} are divided by $\delta = 32$ to produce the final disparity estimations of the ground-truth light fields Ψ_e^2 ($1 \leqslant e \leqslant 5$), which are shown in the left part of Table II. It can be seen that the d_{range}^{Ψ} value of Ψ_1^2 is above 2 pixels, the d_{range}^{Ψ} values of Ψ_2^2 and Ψ_5^2 are between 1 - 2 pixels, and the d_{range}^{Ψ} values of Ψ_3^2 and Ψ_4^2 are less than 1 pixel. Since the baseline approaches (SepConv and PIASC) require the interpolation rate δ^2 to be a power of two and DRST requires that $d_{range}^{\mathcal{S}} = (d_{range}^{\Psi} \cdot \delta^2) \leqslant \tau\ (= 16)$ for any input SSLF, the interpolation rate δ^2 is set to 4 for Ψ_1^2, 8 for Ψ_2^2 and Ψ_5^2, and 16 for Ψ_3^2 and Ψ_4^2, respectively. Therefore, five input SSLFs $\mathcal{S}_e^2$ ($1 \leqslant e \leqslant 5$) are generated. Specifically, $\mathcal{S}_1^2$ has $n^2 = 25$ parallax views, $\mathcal{S}_2^2$ and $\mathcal{S}_5^2$ have $n^2 = 13$ parallax views, and $\mathcal{S}_3^2$ and $\mathcal{S}_4^2$ have $n^2 = 7$ parallax views. The target DSLFs to be reconstructed from these five input SSLFs have varying angular resolutions. To be precise, $\mathcal{D}_1^2$ has $\dot{n}^2 = 385$ parallax views, $\mathcal{D}_2^2$ and $\mathcal{D}_5^2$ have $\dot{n}^2 = 193$ parallax views, and $\mathcal{D}_3^2$ and $\mathcal{D}_4^2$ have $\dot{n}^2 = 97$ parallax views.

4) Evaluation Dataset 3: The evaluation dataset 3 is Centre for Immersive Visual Technologies (CIVIT) DSLF dataset, which was prepared for IEEE International Conference on Multimedia and Expo (ICME) 2018 grand challenge on DSLF reconstruction [18, 35]. The dataset has five real-world horizontal-parallax light fields. In particular, the evaluation dataset 3 contains five ground-truth horizontal-parallax light fields Ψ_e^3 ($1 \leqslant e \leqslant 5$) with the same angular and spatial resolutions, *i.e.* $\ddot{n}^3 = 193$, $l^3 = 720$ and $m^3 = 1280$. The middle views of these ground-truth light fields are illustrated in Fig. 6. The input SSLFs $\mathcal{S}_e^3$ ($1 \leqslant e \leqslant 5$) of the evaluation dataset 3 are generated from Ψ_e^3 ($1 \leqslant e \leqslant 5$) using the interpolation rate $\delta^3 = 16$. Each generated SSLF has $n^3 = 13$ parallax images accordingly. The disparity data of all the input SSLFs are estimated manually and exhibited in the left part of Table III. It can be seen that the estimated $d_{range}^{\mathcal{S}}$ values of the input SSLFs meet the aforementioned requirement (b). The target DSLFs $\mathcal{D}_e^3$ ($1 \leqslant e \leqslant 5$) to be reconstructed from the input SSLFs have the same angular resolution as the ground-truth light fields Ψ_e^3 ($1 \leqslant e \leqslant 5$), *i.e.* $\dot{n}^3 = \ddot{n}^3 = 193$.

(a) Ψ_1^3: Seal and balls (b) Ψ_2^3: Castle (c) Ψ_3^3: Holiday (d) Ψ_4^3: Dragon (f) Ψ_5^3: Flowers

Figure 6. Middle views of ground-truth light fields Ψ_e^3 ($1 \leqslant e \leqslant 5$) in the evaluation dataset 3.

Table III
DISPARITY ESTIMATION, MINIMUM AND AVERAGE PER-VIEW PSNR RESULTS (IN DB, EXPLAINED IN SECTION IV-B1) FOR THE PERFORMANCE EVALUATION OF DIFFERENT LIGHT FIELD RECONSTRUCTION METHODS ON THE EVALUATION DATASET 3.

e of $\mathcal{S}_e^3$	Disparity Estimation (pixel)			Minimum PSNR / Average PSNR (dB)		
	$d^{\mathcal{S}}_{min}$	$d^{\mathcal{S}}_{max}$	$d^{\mathcal{S}}_{range}$	SepConv ($\mathcal{L}_1$) [17]	ST [14]	DRST
1	-10.5	3.5	14	41.619 / 43.194	37.473 / 42.883	**43.051 / 44.080**
2	-2	12	14	**35.256 / 37.167**	34.719 / 36.969	35.253 / 37.141
3	-8	6	14	**30.884 / 34.709**	30.428 / 34.017	30.631 / 34.363
4	-9	7	16	41.061 / 41.808	38.087 / 41.876	**41.389 / 42.429**
5	-6.5	7.5	14	36.018 / **37.994**	34.851 / 37.787	**36.186** / 37.857

5) Implementation details: The proposed DRST approach is implemented using TensorFlow 2[1] and trained on a server with an Nvidia GeForce RTX 2080 Ti GPU for ten epochs. The optimization tool for minimizing the loss function (4) is AdaMax [36] with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. In addition, the learning rate is set to 10^{-3} for the first two epochs and then adjusted to 10^{-4} for the rest eight epochs. A mini-batch is composed of four sparsely-sampled EPIs $\hat{\varepsilon}$ that are extracted from $\ddot{\varepsilon}$ as described in Section III-C (see Fig. 2 (d) and (c)). Considering that each $\hat{\varepsilon}$ has three color channels, a mini-batch actually comprises 12 one-channel EPIs. Since the number of training samples is given above in Section IV-A1, each epoch has 124,416 training iterations. The encoder-decoder network $\mathcal{R}$ in Section III-D has $3,618,959$ trainable parameters. It takes around 32 hours to finish the whole training process. Regarding the evaluation on the above three evaluation datasets, all the methods are conducted on a local machine with an Nvidia GeForce RTX 2070 GPU. It should be noted that when evaluating DRST on an input evaluation SSLF with an angular resolution n (> 3), this SSLF requires to be converted into $\lfloor \frac{n}{2} \rfloor$ sub-SSLFs, of which each has the same angular resolution, *i.e.* 3 pixels. The parameters of ST using the DORE algorithm are set in accordance with [14], *i.e.* $\alpha = 20$ with 100 iterations and a low-pass initial estimation for each input sparsely-sampled EPI.

B. Results and Analysis

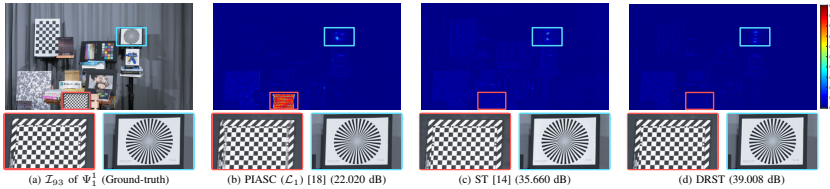
All the light field reconstruction methods are evaluated quantitatively and qualitatively as bellow.

1) Quantitative evaluation: The minimum and average per-view PSNRs between the reconstructed DSLF $\mathcal{D}$ and ground-truth light field Ψ are utilized to evaluate the light field reconstruction performance. The quantitative evaluation results of DRST and the other three state-of-the-art light field reconstruction methods on the aforementioned three evaluation datasets are presented in Table I, Table II and Table III. Looking at the DSLF reconstruction results in Table I, it is apparent that DRST outperforms the other three methods *w.r.t.* minimal PSNR on all the input SSLFs of the evaluation dataset 1 except for $\mathcal{S}_3^1$. It is noticeable that on $\mathcal{S}_1^1$ and $\mathcal{S}_6^1$, the minimal PSNR results of DRST are 2.964 and 3.825 dB higher than those of the second-best method, *i.e.* ST. Regarding the average PSNR results, DRST is better than ST on $\mathcal{S}_e^1$, $e \in \{1, 6, 7, 8\}$ and comparable to ST on $\mathcal{S}_2^1$ and $\mathcal{S}_4^1$; however, on the rest three input SSLFs, ST achieves better performance than DRST. Moreover, both DRST and ST significantly outperform PIASC and SepConv *w.r.t.* both minimum and average PSNRs on all the input SSLFs of the evaluation dataset 1. The main reason for this is that the light field scenes of the evaluation dataset 1 have repetitive patterns that can hardly be handled by video frame interpolation-based methods, since they are incapable of knowing the context information, *i.e.* the moving direction and speed of the virtual camera. In addition, PIASC and SepConv have almost the same performance on all the input SSLFs of the evaluation dataset 1, implying that the fine-tuning strategy of PIASC helps little in improving the performance of SepConv on the evaluation dataset 1.

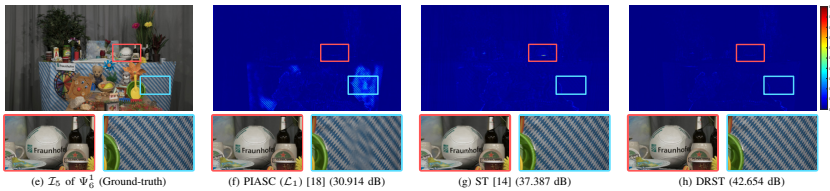
The minimum and average PSNRs of all the light field reconstruction methods on the evaluation dataset 2 are compared in Table II. With regard to minimum PSNR, the proposed DRST performs better than the second-best method, *i.e.* PIASC, on $\mathcal{S}_e^2$, $e \in \{2, 3, 5\}$ and comparably to PIASC on $\mathcal{S}_1^2$, which demonstrates the effectiveness of DRST for DSLF reconstruction in real-world environments. It can also be found that on $\mathcal{S}_5^2$, the minimum PSNR value of DRST is 2.248 dB higher than that of PIASC. In terms of average PSNR, PIASC achieves the best results among all the four light field reconstruction methods, implying that the video frame interpolation-based methods can better handle the DSLF reconstruction for light field scenes without repetitive patterns. Moreover, ST performs worst among all the light field reconstruction methods *w.r.t.* both minimum and average PSNRs. Furthermore, the performance of PIASC is slightly better than that of SepConv on all input SSLFs in terms of both minimum and average PSNRs, indicating that the the fine-tuning strategy of PIASC is effective in improving the performance of SepConv for the real-world light field scenes of the evaluation dataset 2.

The quantitative results of three light field reconstruction methods on the evaluation dataset 3 are compared in Table III.

[1]https://github.com/ygaostu/DRST (to appear)

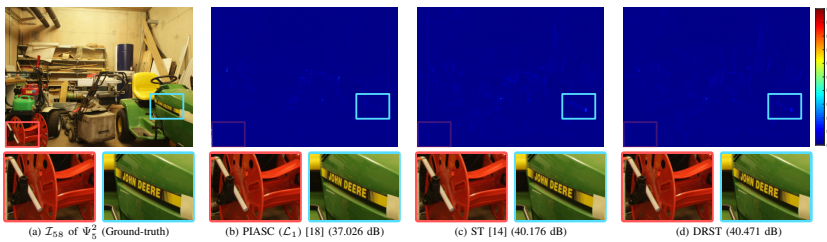


(a) $\mathcal{I}_{93}$ of Ψ_1^1 (Ground-truth) (b) PIASC ($\mathcal{L}_1$) [18] (22.020 dB) (c) ST [14] (35.660 dB) (d) DRST (39.008 dB)



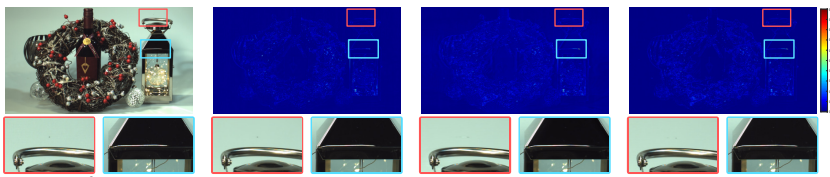
(e) $\mathcal{I}_5$ of Ψ_6^1 (Ground-truth) (f) PIASC ($\mathcal{L}_1$) [18] (30.914 dB) (g) ST [14] (37.387 dB) (h) DRST (42.654 dB)

Figure 7. Light field reconstruction results on the evaluation dataset 1.



(a) $\mathcal{I}_{58}$ of Ψ_5^2 (Ground-truth) (b) PIASC ($\mathcal{L}_1$) [18] (37.026 dB) (c) ST [14] (40.176 dB) (d) DRST (40.471 dB)

Figure 8. Light field reconstruction results on the evaluation dataset 2.



(a) $\mathcal{I}_6$ of Ψ_3^3 (Ground-truth) (b) SepConv ($\mathcal{L}_1$) [17] (33.479 dB) (c) ST [14] (31.315 dB) (d) DRST (32.708 dB)

Figure 9. Light field reconstruction results on the evaluation dataset 3.

The results of PIASC are omitted in this table for two reasons: (i) PIASC is an enhanced SepConv that is fine-tuned on the ground-truth light fields Ψ_1^3, Ψ_2^3 and Ψ_3^3 of the evaluation dataset 3, since these three light fields were the training data provided for the ICME grand challenge; (ii) the learning-based SepConv and DRST are neither trained nor fine-turned on the evaluation dataset 3. It can be seen from the minimum PSNR data in the table that DRST achieves the best performance on three input SSLFs $\mathcal{S}_e^3$, $e \in \{1, 4, 5\}$. Besides, DRST is comparable to SepConv on $\mathcal{S}_2^3$. It can also be found that the minimum PSNR of DRST is 1.432 dB higher than that of SepConv on $\mathcal{S}_1^3$. As regards average PSNR, DRST performs best on $\mathcal{S}_1^3$ and $\mathcal{S}_4^3$. Furthermore, in terms of both minimum and average PSNRs, the performance of DRST is better than that of ST, demonstrating the superiority of DRST over ST.

2) Qualitative evaluation: The qualitative evaluation results of three light field reconstruction methods on the evaluation dataset 1 are illustrated in Fig. 7. Since SepConv and PIASC perform almost the same as discussed above, the results of SepConv are skipped here. The top row exhibits the reconstruction results corresponding to $\mathcal{I}_{93}$ of Ψ_1^1. The checkerboard and Siemens star are chosen as the interesting areas. As shown in the figure (b), PIASC fails in reconstructing the checkerboard, because this algorithm can hardly exploit the context information; in other words, the moving direction and speed of the checkerboard are unknown to it. However, ST and

Table IV
THE AVERAGE COMPUTATION TIME (MS) OF DENSELY-SAMPLED EPI RECONSTRUCTION ON A COLOR SPARSELY-SAMPLED EPI ε.

Size of input ε (pixels)	ST [14]	DRST	Speedup
$1280 \times 13 \times 3$	1529.1	640.5	2.4x
$960 \times 25 \times 3$	3258.9	1072.1	3.0x
$960 \times 13 \times 3$	1792.7	533.7	3.4x
$960 \times 7 \times 3$	1125.2	236.9	4.7x

DRST do not have such problem. The reconstructed checkerboards are shown in (c) and (d), respectively. Regarding the Siemens star, all these three methods have small artifacts. The bottom row shows the reconstruction results *w.r.t.* $\mathcal{I}_5$ of Ψ_6^1. The two interesting areas are the Fraunhofer-logo ball and table curtain with repetitive pattern. As shown in (f), PIASC fails in reconstructing the table curtain, since the context information is unavailable to it. However, ST and DRST overcomes this problem by leveraging the context information implicitly encoded by EPIs. Their results are presented in (g) and (h), respectively. As regards the Fraunhofer logo, ST produces small artifacts when reconstructing the letters, while PIASC and DRST generate visually-correct results.

The visualized light filed reconstruction results on the evaluation dataset 2 are illustrated in Fig. 8. The results of SepConv are omitted here because the SepConv-based PIASC works slightly better than SepConv on the evaluation dataset 2 as discussed above. The reconstructed results corresponding to $\mathcal{I}_{58}$ of Ψ_5^2 are compared in this figure. As shown in the red box of (b), PIASC fails in reconstructing the left border correctly. However, ST and DRST recover the left border with visually-correct results as shown in (c) and (d), respectively. Regarding the reconstruction of the vertical bars close to the right side of the John Deere logo, both ST and DRST have blurry artifacts; nevertheless, PIASC achieves sharp results for the recovery of these vertical bars.

The light field reconstruction results of three different methods on the evaluation dataset 3 are illustrated in Fig. 9. The $\mathcal{I}_6$ of Ψ_3^3 is chosen to be the reference. The red and blue blocks in (a) denote two interesting areas. The red-block interesting area contains the background of the light field scene, which is a flat ground with a black dot. The blue-block interesting area has a horizontal shiny line on the metal frame of the lantern. As shown in (b), SepConv succeeds in reconstructing the black dot and the shiny line with visually-correct results. However, for both cases, ST and DRST fail in generating visually-correct results. Specifically, two blurry black dots appear in both (c) and (d), because the background floor in the real-world light field scenes of the evaluation dataset 3 is out of the disparity range that ST and DRST are designed to handle. Besides, the shiny line is extended in both (c) and (d), because both ST and DRST are designed to handle DSLF reconstruction for Lambertian scenes or non-Lambertian scenes consisting of semi-transparent objects only, while this shiny line is on a non-Lambertian reflection surface.

3) Computation time: In addition to the above evaluations suggesting that DRST is more effective than ST, the computation time of ST and DRST for densely-sampled EPI reconstruction on the input sparsely-sampled EPIs with varying sizes in the aforementioned three evaluation datasets is compared in Table IV. As can be seen from this table, the proposed DRST is at least 2.4 times faster than ST, since DRST performs the shearlet domain transformations for only one iteration. Besides, looking at the data of rows one and three, where the angular resolutions of the input sparsely-sampled EPIs are the same, DRST achieves a higher speedup over ST for the input EPI with a lower spatial resolution, *i.e*. 960 pixels. Moreover, it can be seen from the data of rows two, three and four that for the same spatial resolution of the input SSLF, the speedup of DRST over ST gets higher when the angular resolution of the input sparsely-sampled EPIs gets smaller, *i.e*. from 25 to 13, then to 7 pixels. In summary, the value of the speedup of DRST over ST depends on the size of the input sparsely-sampled EPI, *i.e*. the speedup value will be higher if the size of the input sparsely-sampled EPI is smaller.

V. CONCLUSION

This paper has presented a novel learning-based method, DRST, for DSLF reconstruction on SSLFs with disparity ranges up to 16 pixels. The proposed DRST takes advantage of a deep CNN, consisting of an encoder-decoder network and a residual learning strategy, to perform sparse regularization in the shearlet transform domain of an input sparsely-sampled EPI, thereby fulfilling image inpainting on this EPI in its image domain. The end-to-end fully convolutional network of DRST is trained on synthetic SSLF data only by leveraging the elaborately-designed masks. Experimental results on three different challenging evaluation datasets consisting of real-world light field scenes with varying moderate disparity ranges (8 - 16 pixels) show that the learning-based DRST performs better than the non-learning-based ST and comparably to the other state-of-the-art light field reconstruction methods. Moreover, DRST is a time-efficient algorithm that is at least 2.4 times faster than ST.

REFERENCES

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*, 1996, pp. 31–42. 1

[2] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Image based rendering technique via sparse representation in shearlet domain," in *ICIP*, 2015, pp. 1379–1383. 1

[3] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field image processing: An overview," *IEEE J-STSP*, vol. 11, no. 7, pp. 926–954, 2017. 1

[4] A. Smolic, "3D video and free viewpoint video - from capture to display," *Pattern Recognition*, vol. 44, no. 9, pp. 1958–1968, 2011. 1

[5] J. Yu, "A light-field journey to virtual reality," *IEEE MultiMedia*, vol. 24, no. 2, pp. 104–112, 2017. 1

[6] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," *Computer Science Technical Report CSTR*, vol. 2, no. 11, pp. 1–11, 2005. 1

[7] C. Perwaß and L. Wietzke, "Single lens 3D-camera with extended depth-of-field," in *Human Vision and Electronic Imaging XVII*, vol. 8291, 2012, pp. 45 – 59. 1

[8] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," in *ACM TOG*, vol. 24, no. 3, 2005, pp. 765–776. 1

[9] J. Flynn, M. Broxton, P. Debevec, M. DuVall, G. Fyffe, R. Overbeck, N. Snavely, and R. Tucker, "Deepview: View synthesis with learned gradient descent," in *CVPR*, 2019, pp. 2367–2376. 1

[10] S. Lu, "High-speed video from asynchronous camera array," in *WACV*, 2019, pp. 2196–2205. 1

[11] S. Babacan, R. Ansorge, M. Luessi, P. Mataran, R. Molina, and A. Katsaggelos, "Compressive light field sensing," *IEEE TIP*, vol. 21, no. 12, pp. 4746–4757, 2012. 1
[12] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, "Compressive light field photography using overcomplete dictionaries and optimized projections," *ACM TOG*, vol. 32, no. 4, p. 46, 2013. 1
[13] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE TPAMI*, vol. 40, no. 1, pp. 133–147, 2018. 1, 2, 3
[14] ——, "Accelerated shearlet-domain light field reconstruction," *IEEE J-STSP*, vol. 11, no. 7, pp. 1082–1091, 2017. 1, 2, 5, 6, 7, 8, 9
[15] H.-Y. Shum, S.-C. Chan, and S.-B. Kang, *Image-based rendering*. Springer Science+Business Media, 2007. 1
[16] H.-Y. Shum, S.-B. Kang, and S.-C. Chan, "Survey of image-based representations and compression techniques," *IEEE TCSVT*, vol. 13, no. 11, pp. 1020–1037, 2003. 1
[17] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *ICCV*, 2017, pp. 261–270. 2, 5, 6, 7, 8
[18] Y. Gao and R. Koch, "Parallax view generation for static scenes using parallax-interpolation adaptive separable convolution," in *ICME Workshops*, 2018, pp. 1–4. 2, 5, 6, 8
[19] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM TOG*, vol. 35, no. 6, pp. 193:1–193:10, 2016. 2
[20] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, "Light field reconstruction using deep convolutional network on EPI," in *CVPR*, 2017, pp. 1638–1646. 2
[21] H. Yeung, J. Hou, J. Chen, Y. Chung, and X. Chen, "Fast light field reconstruction with deep coarse-to-fine modelling of spatial-angular clues," in *ECCV*, 2018, pp. 137–152. 2
[22] Y. Wang, F. Liu, Z. Wang, G. Hou, Z. Sun, and T. Tan, "End-to-end view synthesis for light field imaging with pseudo 4DCNN," in *ECCV*, 2018, pp. 340–355. 2
[23] Y. Gao, R. Bregovic, A. Gotchev, and R. Koch, "MAST: Mask-accelerated shearlet transform for densely-sampled light field reconstruction," in *ICME*, 2019, pp. 187–192. 2
[24] G. Kutyniok, W.-Q. Lim, and R. Reisenhofer, "Shearlab 3D: Faithful digital shearlet transforms based on compactly supported shearlets," *ACM Transactions on Mathematical Software (TOMS)*, vol. 42, no. 1, 2016. 2
[25] G. Kutyniok and D. Labate, *Shearlets: Multiscale analysis for multivariate data*. Springer Science+Business Media, 2012. 2
[26] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4D light fields," in *ACCV*, 2016, pp. 19–34. 3, 5
[27] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015, pp. 234–241. 4
[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778. 4
[29] Y. Gao, R. Koch, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform in tensorflow," in *ICME Workshops*, 2019, pp. 612–612. 5
[30] ——, "IEST: Interpolation-enhanced shearlet transform for light field reconstruction using adaptive separable convolution," in *EUSIPCO*, 2019, pp. 1–5. 5
[31] M. Ziegler, R. op het Veld, J. Keinert, and F. Zilly, "Acquisition system for dense lightfield of large scenes," in *3DTV-CON*, 2017, pp. 1–4. 5
[32] A. Gotchev, *Spline and wavelet based techniques for signal and image processing*. Thesis for the degree of Doctor of Technology, Tampere University of Technology (Tampere, Finland), 2003. 6
[33] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *CVPR*, 2018, pp. 8934–8943. 6
[34] V. K. Adhikarla, M. Vinkler, D. Sumin, R. K. Mantiuk, K. Myszkowski, H.-P. Seidel, and P. Didyk, "Towards a quality metric for dense light fields," in *CVPR*, 2017, pp. 58–67. 6
[35] S. Moreschini, F. Gama, R. Bregovic, and A. Gotchev, "CIVIT dataset: Horizontal-parallax-only densely-sampled light-fields," in *European Light Field Imaging Workshop*, 2019. 6
[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. 7

Bibliography

[AAF+14] H. Afzal, D. Aouada, D. Font, B. Mirbach, and B. Ottersten. "RGB-D multi-view system calibration for full 3D scene reconstruction". In: *International Conf. on Pattern Recognition (ICPR)*. 2014, pp. 2459–2464.

[ABF+06] T. Agocs, T. Balogh, T. Forgacs, F. Bettio, E. Gobbetti, G. Zanetti, and E. Bouvier. "A large scale interactive holographic display". In: *IEEE Virtual Reality Conf. (VR)*. 2006, pp. 311–311.

[AHB87] K. S. Arun, T. S. Huang, and S. D. Blostein. "Least-squares fitting of two 3-D point sets". In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* PAMI-9.5 (1987), pp. 698–700.

[Alt92] N. S. Altman. "An introduction to kernel and nearest-neighbor nonparametric regression". In: *The American Statistician* 46.3 (1992), pp. 175–185.

[AN10] A. Ashok and M. A. Neifeld. "Compressive light field imaging". In: *Three-Dimensional Imaging, Visualization, and Display 2010 and Display Technologies and Applications for Defense, Security, and Avionics IV*. Vol. 7690. SPIE, 2010, pp. 221–232.

[API19] J. An, S. Park, and I. Ihm. "Construction of a flexible and scalable 4D light field camera array using Raspberry Pi clusters". In: *The Visual Computer* 35.10 (2019), pp. 1475–1488.

[ASS+10] S. Agarwal, N. Snavely, S. M. Seitz, and R. Szeliski. "Bundle adjustment in the large". In: *European Conf. on Computer Vision (ECCV)*. 2010, pp. 29–42.

[BAL+12] S. D. Babacan, R. Ansorge, M. Luessi, P.R. Mataran, R. Molina, and A.K. Katsaggelos. "Compressive light field sensing". In: *IEEE Tran. on Image Processing (TIP)* 21.12 (2012), pp. 4746–4757.

[Bal06] T. Balogh. "The HoloVizio system". In: *Stereoscopic Displays and Virtual Reality Systems XIII*. Vol. 6055. SPIE, 2006, pp. 279–290.

[Ben75] J. L. Bentley. "Multidimensional binary search trees used for associative searching". In: *Communications of the ACM* 18.9 (1975), pp. 509–517.

[BLM+19] W. Bao, W.-S. Lai, C. Ma, X. Zhang, Z. Gao, and M.-H. Yang. "Depth-aware video frame interpolation". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 3703–3712.

[BLZ+19] W. Bao, W.-S. Lai, X. Zhang, Z. Gao, and M.-H. Yang. "MEMC-Net: motion estimation and motion compensation driven neural network for video interpolation and enhancement". In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* (2019).

[BM92] P. J. Besl and N. D. McKay. "A method for registration of 3-D shapes". In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* 14.2 (1992), pp. 239–256.

[Bro71] D. C. Brown. "Close-range camera calibration". In: *Photogrammetric Engineering* 37.8 (1971), pp. 855–866.

[BSV+19] R. Bregovic, E. Sahin, S. Vagharshakyan, and A. Gotchev. "Signal processing methods for light field displays". In: *Handbook of Signal Processing Systems*. Springer International Publishing, 2019, pp. 3–50.

[BTV06] H. Bay, T. Tuytelaars, and L. Van Gool. "SURF: speeded up robust features". In: *European Conf. on Computer Vision (ECCV)*. 2006, pp. 404–417.

[CGM+16] A. Corti, S. Giancola, G. Mainetti, and R. Sala. "A metrological characterization of the Kinect V2 time-of-flight camera". In: *Robotics and Autonomous Systems* 75 (2016), pp. 584–594.

[CL96] B. Curless and M. Levoy. "A volumetric method for building complex models from range images". In: *ACM SIGGRAPH*. 1996, pp. 303–312.

[CSG+15] A. Chuchvara, O. Suominen, M. Georgiev, and A. Gotchev. "Speed-optimized free-viewpoint rendering based on depth layering". In: *3DTV-CON*. 2015, pp. 1–4.

[CSN07] S. C. Chan, H.-Y. Shum, and K.-T. Ng. "Image-based rendering and synthesis". In: *IEEE Signal Processing Magazine* 24.6 (2007), pp. 22–33.

[CTC+00] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum. "Plenoptic sampling". In: *ACM SIGGRAPH*. 2000, pp. 307–318.

[CTJ+17] D.-M. Córdova-Esparza, J. R. Terven, H. Jiménez-Hernández, and A.-M. Herrera-Navarro. "A multiple camera calibration and point cloud fusion tool for Kinect V2". In: *Science of Computer Programming* 143 (2017), pp. 1–8.

[DFB+19] M. DuVall, J. Flynn, M. Broxton, and P. Debevec. "Compositing light field video using multiplane images". In: *ACM SIGGRAPH 2019 Posters*. 2019, pp. 1–2.

[DZD06] A. L. Da Cunha, J. Zhou, and M. N Do. "The nonsubsampled contourlet transform: theory, design, and applications". In: *IEEE Tran. on Image Processing (TIP)* 15.10 (2006), pp. 3089–3101.

[EGM+16] S. Esquivel, Y. Gao, T. Michels, L. Palmieri, and R. Koch. "Synchronized data capture and calibration of a large-field-of-view moving multi-camera light field rig". In: *3DTV-CON Workshops*. 2016.

[FB81] M. A. Fischler and R. C. Bolles. "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography". In: *Communications of the ACM* 24.6 (1981), pp. 381–395.

[FBD+19] J. Flynn, M. Broxton, P. Debevec, M. DuVall, G. Fyffe, R. Overbeck, N. Snavely, and R. Tucker. "DeepView: view synthesis with learned gradient descent". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 2367–2376.

[FDP06] C. Fehn, R. De La Barré, and S. Pastoor. "Interactive 3-DTV-concepts and key technologies". In: *Proceedings of the IEEE* 94.3 (2006), pp. 524–538.

[FEF+17] D. Fourure, R. Emonet, E. Fromont, D. Muselet, A. Trémeau, and C. Wolf. "Residual Conv-Deconv grid network for semantic segmentation". In: *British Machine Vision Conf. (BMVC)*. 2017.

[Feh04] C. Fehn. "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV". In: *Stereoscopic Displays and Virtual Reality Systems XI*. Vol. 5291. SPIE, 2004, pp. 93–104.

[FL15] S. A. Fezza and M.-C. Larabi. "Color correction for stereo and multi-view coding". In: *Color Image and Video Enhancement*. Springer, 2015, pp. 291–314.

[FMT+06] T. Fujii, K. Mori, K. Takeda, K. Mase, M. Tanimoto, and Y. Suenaga. "Multipoint measuring system for video and sound - 100-camera and microphone system". In: *International Conf. on Multimedia and Expo (ICME)*. 2006, pp. 437–440.

[FTV00] A. Fusiello, E. Trucco, and A. Verri. "A compact algorithm for rectification of stereo pairs". In: *Machine Vision and Applications* 12.1 (2000), pp. 16–22.

[GBG+19] Y. Gao, R. Bregovic, A. Gotchev, and R. Koch. "MAST: mask-accelerated shearlet transform for densely-sampled light field reconstruction". In: *International Conf. on Multimedia and Expo (ICME)*. 2019, pp. 187–192.

[GBG20] Y. Gao, R. Bregovic, and A. Gotchev. "Self-supervised light field reconstruction using shearlet transform and cycle consistency". In: *IEEE Signal Processing Letters (SPL)* (2020).

[GBK+20] Y. Gao, R. Bregovic, R. Koch, and A. Gotchev. "DRST: deep residual shearlet transform for densely-sampled light field reconstruction". In: *arXiv preprint arXiv:2003.08865* (2020).

[GEK+17a] Y. Gao, S. Esquivel, R. Koch, and J. Keinert. "A novel self-calibration method for a stereo-ToF system using a Kinect V2 and two 4K GoPro cameras". In: *International Conf. on 3D Vision (3DV)*. 2017, pp. 201–205.

[GEK+17b] Y. Gao, S. Esquivel, R. Koch, M. Ziegler, F. Zilly, and J. Keinert. "A novel Kinect V2 registration method for large-displacement environments using camera and scene constraints". In: *International Conf. on Image Processing (ICIP)*. 2017, pp. 997–1001.

[GGS+96] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. "The lumigraph". In: *ACM SIGGRAPH*. 1996, pp. 43–54.

[GHT11] S. Gauglitz, T. Höllerer, and M. Turk. "Evaluation of interest point detectors and feature descriptors for visual tracking". In: *International Journal of Computer Vision (IJCV)* 94.3 (2011), pp. 335–360.

[GK14] M. Genzel and G. Kutyniok. "Asymptotic analysis of inpainting via universal shearlet systems". In: *SIAM Journal on Imaging Sciences* 7.4 (2014), pp. 2301–2339.

[GK18] Y. Gao and R. Koch. "Parallax view generation for static scenes using parallax-interpolation adaptive separable convolution". In: *International Conf. on Multimedia and Expo Workshops (ICMEW)*. 2018, pp. 1–4.

[GKB+19a] Y. Gao, R. Koch, R. Bregovic, and A. Gotchev. "FAST: flow-assisted shearlet transform for densely-sampled light field reconstruction". In: *International Conf. on Image Processing (ICIP)*. 2019, pp. 3741–3745.

[GKB+19b] Y. Gao, R. Koch, R. Bregovic, and A. Gotchev. "IEST: interpolation-enhanced shearlet transform for light field reconstruction using adaptive separable convolution". In: *European Signal Processing Conf. (EUSIPCO)*. 2019, pp. 1–5.

[GKB+19c] Y. Gao, R. Koch, R. Bregovic, and A. Gotchev. "Light field reconstruction using shearlet transform in tensorflow". In: *International Conf. on Multimedia and Expo Workshops (ICMEW)*. 2019, pp. 612–612.

[GL10] T. G. Georgiev and A. Lumsdaine. "Focused plenoptic camera and rendering". In: *Journal of Electronic Imaging* 19.2 (2010), pp. 1–11.

[GMK18] Y. Gao, T. Michels, and R. Koch. "A novel Kinect V2 registration method using color and deep geometry descriptors". In: *European Signal Processing Conf. (EUSIPCO)*. 2018, pp. 201–205.

[Got03] A. Gotchev. "Spline and wavelet based techniques for signal and image processing". PhD thesis. Tampere University of Technology, 2003.

[GTK+13] M. Grzegorzek, C. Theobalt, R. Koch, and A. Kolb. *Time-of-flight and depth imaging. sensors, algorithms, and applications*. Springer-Verlag Berlin Heidelberg, 2013.

[GY17] J. Gu and J.C. Ye. "Multi-scale wavelet domain residual learning for limited-angle CT reconstruction". In: *arXiv preprint arXiv:1703.01382* (2017).

[GZZ+17] Y. Gao, M. Ziegler, F. Zilly, S. Esquivel, and R. Koch. "A linear method for recovering the depth of Ultra HD cameras using a Kinect V2 sensor". In: *International Conf. on Machine Vision Applications (MVA)*. 2017, pp. 494–497.

[HHE+16] R. Horaud, M. Hansard, G. Evangelidis, and C. Ménier. "An overview of depth cameras and range scanners based on time-of-flight technologies". In: *Machine Vision and Applications* 27.7 (2016), pp. 1005–1020.

[HJK+16] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke. "A dataset and evaluation methodology for depth estimation on 4D light fields". In: *Asian Conf. on Computer Vision (ACCV)*. 2016, pp. 19–34.

[Hor87] B. K. P. Horn. "Closed-form solution of absolute orientation using unit quaternions". In: *JOSA A* 4.4 (1987), pp. 629–642.

[HR17] J. Hur and S. Roth. "MirrorFlow: exploiting symmetries in joint optical flow and occlusion estimation". In: *International Conf. on Computer Vision (ICCV)*. 2017, pp. 312–321.

[HZ03] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[HZR+16] K. He, X. Zhang, S. Ren, and J. Sun. “Deep residual learning for image recognition”. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778.

[IMS+17] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. “FlowNet 2.0: evolution of optical flow estimation with deep networks”. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1647–1655.

[IW05] A. Ilie and G. Welch. “Ensuring color consistency across multiple cameras”. In: *International Conf. on Computer Vision (ICCV)*. Vol. 2. 2005, pp. 1268–1275.

[IZZ+17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. “Image-to-image translation with conditional adversarial networks”. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1125–1134.

[JMA06] N. Joshi, W. Matusik, and S. Avidan. “Natural video matting using camera arrays”. In: *ACM Tran. on Graphics (TOG)* 25.3 (2006), pp. 779–786.

[Jos04] N. S. Joshi. *Color calibration for arrays of inexpensive image sensors*. Tech. rep. 2004-02. Department of Computer Science, Stanford University, 2004, pp. 1–26.

[JSJ+18] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz. “Super SloMo: high quality estimation of multiple intermediate frames for video interpolation”. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 9000–9008.

[JT20] L. Jing and Y. Tian. “Self-supervised visual feature learning with deep neural networks: a survey”. In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* (2020).

[KBK+10] A. Kolb, E. Barth, R. Koch, and R. Larsen. “Time-of-flight cameras in computer graphics”. In: *Computer Graphics Forum (CGF)* 29.1 (2010), pp. 141–159.

[KHP+99] R. Koch, B. Heigl, M. Pollefeys, L. Van Gool, and H. Niemann. "A geometric approach to light field calibration". In: *International Conf. on Computer Analysis of Images and Patterns (CAIP)*. Springer. 1999, pp. 596–603.

[KHP01] R. Koch, B. Heigl, and M. Pollefeys. "Image-based rendering from uncalibrated lightfields with scalable geometry". In: *Multi-Image Analysis*. Springer, 2001, pp. 51–66.

[KL12] G. Kutyniok and D. Labate. *Shearlets: multiscale analysis for multivariate data*. Springer Science+Business Media, 2012.

[KLR16] G. Kutyniok, W.-Q. Lim, and R. Reisenhofer. "Shearlab 3D: faithful digital shearlet transforms based on compactly supported shearlets". In: *ACM Tran. on Mathematical Software (TOMS)* 42.5:1–5:42 (2016).

[KND15] M. Kowalski, J. Naruniec, and M. Daniluk. "LiveScan3D: a fast and inexpensive 3D data acquisition system for multiple Kinect v2 sensors". In: *International Conf. on 3D Vision (3DV)*. 2015, pp. 318–325.

[KSZ12] G. Kutyniok, M. Shahram, and X. Zhuang. "Shearlab: a rational design of a digital parabolic scaling algorithm". In: *SIAM Journal on Imaging Sciences* 5.4 (2012), pp. 1291–1332.

[KWR16] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi. "Learning-based view synthesis for light field cameras". In: *ACM Tran. on Graphics (TOG)* 35.6 (2016), 193:1–193:10.

[KZP+13] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. Gross. "Scene reconstruction from high spatio-angular resolution light fields". In: *ACM Tran. on Graphics (TOG)* 32.4 (2013), 73:1–73:12.

[LA09] M. I. A. Lourakis and A. A. Argyros. "SBA: a software package for generic sparse bundle adjustment". In: *ACM Tran. on Mathematical Software (TOMS)* 36.1 (2009), 2:1–2:30.

[LC87] W. E. Lorensen and H. E. Cline. "Marching cubes: a high resolution 3D surface construction algorithm". In: *ACM SIGGRAPH*. Vol. 21. 4. 1987, pp. 163–169.

[LDX11] K. Li, Q. Dai, and W. Xu. “Collaborative color calibration for multi-camera systems”. In: *Signal Processing: Image Communication* 26.1 (2011), pp. 48–60.

[LG09] A. Lumsdaine and T. Georgiev. “The focused plenoptic camera”. In: *International Conf. on Computational Photography (ICCP)*. 2009, pp. 1–8.

[LH96] M. Levoy and P. Hanrahan. “Light field rendering”. In: *ACM SIGGRAPH*. 1996, pp. 31–42.

[LHM00] C.-P. Lu, G. D. Hager, and E. Mjolsness. “Fast and globally convergent pose estimation from video images”. In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* 22.6 (2000), pp. 610–622.

[LLS+13] H. Lakshman, W.-Q. Lim, H. Schwarz, D. Marpe, G. Kutyniok, and T. Wiegand. “Image interpolation using shearlet based sparsity priors”. In: *International Conf. on Image Processing (ICIP)*. 2013, pp. 655–659.

[LLW+08] C.-K. Liang, T.-H. Lin, B.-Y. Wong, C. Liu, and H. H. Chen. “Programmable aperture photography: multiplexed light field acquisition”. In: *ACM Tran. on Graphics (TOG)* 27.3 (2008), pp. 1–10.

[LMF09] V. Lepetit, F. Moreno-Noguer, and P. Fua. “EPnP: an accurate O(n) solution to the PnP problem”. In: *International Journal of Computer Vision (IJCV)* 81.2 (2009), pp. 155–166.

[LMH+18] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. “Noise2noise: learning image restoration without clean data”. In: *International Conf. on Machine Learning (ICML)*. 2018, pp. 2965–2974.

[Low04a] K.-L. Low. *Linear least-squares optimization for point-to-plane ICP surface registration*. Tech. rep. 04-004. Department of Computer Science, University of North Carolina at Chapel Hill, 2004, pp. 1–3.

[Low04b] D. G. Lowe. “Distinctive image features from scale-invariant keypoints”. In: *International Journal of Computer Vision (IJCV)* 60.2 (2004), pp. 91–110.

[LQF+16] G. Lafruit, S. Quackenbush, S. Foessel, and A. Hinds. "Technical report of the joint ad hoc group for digital representations of light/sound fields for immersive media applications". In: *meeting of MPEG 115 - Geneva* (2016).

[LSK+10] M. Lindner, I. Schiller, A. Kolb, and R. Koch. "Time-of-flight sensor calibration for accurate range sensing". In: *Computer Vision and Image Understanding (CVIU)* 114.12 (2010), pp. 1318–1328.

[LXD+20] Z. Li, W. Xian, A. Davis, and N. Snavely. "Crowdsampling the plenoptic function". In: *European Conf. on Computer Vision (ECCV)*. 2020.

[LXP+19] S. Li, X. Xu, Z. Pan, and W. Sun. "Quadratic video interpolation for VTSR challenge". In: *International Conf. on Computer Vision Workshops (ICCVW)*. 2019, pp. 3427–3431.

[LXX12] S. Li, C. Xu, and M. Xie. "A robust O(n) solution to the perspective-n-point problem". In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* 34.7 (2012), pp. 1444–1450.

[LYT+17] Z. Liu, R.A. Yeh, X. Tang, Y. Liu, and A. Agarwala. "Video frame synthesis using deep voxel flow". In: *International Conf. on Computer Vision (ICCV)*. 2017, pp. 4473–4481.

[MBB08] Z. Megyesi, A. Barsi, and T. Balogh. "3D video visualization on the holovizio system". In: *3DTV-CON*. 2008, pp. 269–272.

[MDM+18] S. Meyer, A. Djelouah, B. McWilliams, A. Sorkine-Hornung, M. Gross, and C. Schroers. "PhaseNet for video frame interpolation". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 498–507.

[MHR18] S. Meister, J. Hur, and S. Roth. "UnFlow: unsupervised learning of optical flow with a bidirectional census loss". In: *Association for the Advancement of Artificial Intelligence (AAAI)*. 2018.

[MP04] W. Matusik and H. Pfister. "3D TV: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes". In: *ACM Tran. on Graphics (TOG)* 23.3 (2004), pp. 814–824.

[MRS+20] R. Martin-Brualla, N. Radwan, M. S. M. Sajjadi, J. T. Barron, A. Dosovitskiy, and D. Duckworth. "Nerf in the wild: neural radiance fields for unconstrained photo collections". In: *arXiv preprint arXiv:2008.02268* (2020).

[MSG17] T. Milliron, C. Szczupak, and O. Green. "Hallelujah: the world's first lytro VR experience". In: *ACM SIGGRAPH 2017 VR Village*. 2017, 7:1–7:2.

[MSO+19] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar. "Local light field fusion: practical view synthesis with prescriptive sampling guidelines". In: *ACM Tran. on Graphics (TOG)* 38.4 (2019), 29:1–29:14.

[MST+20] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. "Nerf: representing scenes as neural radiance fields for view synthesis". In: *European Conf. on Computer Vision (ECCV)*. 2020.

[MWB+13] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar. "Compressive light field photography using overcomplete dictionaries and optimized projections". In: *ACM Tran. on Graphics (TOG)* 32.4 (2013), 46:1–46:12.

[MWZ+15] S. Meyer, O. Wang, H. Zimmer, M. Grosse, and A. Sorkine-Hornung. "Phase-based frame interpolation for video". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1410–1418.

[NDP09] Q. H. Nguyen, M. N. Do, and S. J. Patel. "Depth image-based rendering with low resolution depth". In: *International Conf. on Image Processing (ICIP)*. 2009, pp. 553–556.

[NL18] S. Niklaus and F. Liu. "Context-aware synthesis for video frame interpolation". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 1701–1710.

[NLB+05] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. *Light field photography with a hand-held plenoptic camera*. Tech. rep. 2005-02. Department of Computer Science, Stanford University, 2005, pp. 1–11.

[NML17a] S. Niklaus, L. Mai, and F. Liu. "Video frame interpolation via adaptive convolution". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2270–2279.

[NML17b] S. Niklaus, L. Mai, and F. Liu. "Video frame interpolation via adaptive separable convolution". In: *International Conf. on Computer Vision (ICCV)*. 2017, pp. 261–270.

[NST+19] S. Nah, S. Son, R. Timofte, K. M. Lee, et al. "AIM 2019 challenge on video temporal super-resolution: methods and results". In: *International Conf. on Computer Vision Workshops (ICCVW)*. 2019, pp. 3388–3398.

[OEE+18] R. S. Overbeck, D. Erickson, D. Evangelakos, M. Pharr, and P. Debevec. "A system for acquiring, processing, and rendering panoramic light field stills for virtual reality". In: *ACM Tran. on Graphics (TOG)* 37.6 (2018), pp. 1–15.

[PW12] C. Perwaß and L. Wietzke. "Single lens 3D-camera with extended depth-of-field". In: *Human Vision and Electronic Imaging XVII*. Vol. 8291. SPIE, 2012, pp. 45–59.

[RFB15] O. Ronneberger, P. Fischer, and T. Brox. "U-Net: convolutional networks for biomedical image segmentation". In: *International Conf. on Medical Image Computing and Computer Assisted Intervention (MICCAI)*. 2015, pp. 234–241.

[SBV+17] N. Sabater et al. "Dataset and pipeline for multi-view light-field video". In: *Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2017, pp. 30–40.

[SBZ+18] S. Shekhar, S. K. Beigpour, M. Ziegler, M. Chwesiuk, D. Paleń, K. Myszkowski, J. Keinert, R. Mantiuk, and P. Didyk. "Light-field intrinsic dataset". In: *British Machine Vision Conf. (BMVC)*. 2018.

[Sch66] P. H. Schönemann. "A generalized solution of the orthogonal procrustes problem". In: *Psychometrika* 31.1 (1966), pp. 1–10.

[SCK07] H.-Y. Shum, S.-C. Chan, and S.-B. Kang. *Image-based rendering*. Springer Science+Business Media, 2007.

[SCW+15] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo. "Convolutional LSTM network: a machine learning approach for precipitation nowcasting". In: *Conf. on Neural Information Processing Systems (NeurIPS)*. 2015, pp. 802–810.

[SF95] E.P. Simoncelli and W.T. Freeman. "The steerable pyramid: a flexible architecture for multi-scale derivative computation". In: *International Conf. on Image Processing (ICIP)*. Vol. 3. 1995, pp. 444–447.

[SKC03] H.-Y. Shum, S.-B. Kang, and S.-C. Chan. "Survey of image-based representations and compression techniques". In: *IEEE Tran. on Circuits and Systems for Video Technology (TCSVT)* 13.11 (2003), pp. 1020–1037.

[SLK15] H. Sarbolandi, D. Lefloch, and A. Kolb. "Kinect range sensing: structured-light versus time-of-flight Kinect". In: *Computer Vision and Image Understanding (CVIU)* 139 (2015), pp. 1–20.

[SMD+16] V. Soleimani, M. Mirmehdi, D. Damen, S. Hannuna, and M. Camplani. "3D data acquisition and registration using two opposing kinects". In: *International Conf. on 3D Vision (3DV)*. 2016, pp. 128–137.

[Smo11] A. Smolic. "3D video and free viewpoint video - from capture to display". In: *Pattern Recognition* 44.9 (2011), pp. 1958–1968.

[SYL+18] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 8934–8943.

[Tan06] M. Tanimoto. "Overview of free viewpoint television". In: *Signal Processing: Image Communication* 21.6 (2006), pp. 454–461.

[TFT+20] A. Tewari et al. "State of the art on neural rendering". In: *Computer Graphics Forum (CGF)* 39.2 (2020), pp. 701–727.

[TMH+00] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. "Bundle adjustment - A modern synthesis". In: *Vision Algorithms: Theory and Practice*. 2000, pp. 298–372.

[TSL+19] J. Trottnow et al. "The potential of light fields in media productions". In: *ACM SIGGRAPH Asia 2019 Technical Briefs*. 2019, pp. 71–74.

[TTF+11] M. Tanimoto, M.P. Tehrani, T. Fujii, and T. Yendo. "Free-viewpoint TV". In: *IEEE Signal Processing Magazine* 28.1 (2011), pp. 67–76.

[UWH+03] J. Unger, A. Wenger, T. Hawkins, A. Gardner, and P. Debevec. "Capturing and rendering with incident light fields". In: *Eurographics Symposium on Rendering (EGSR)*. 2003, pp. 141–149.

[VA08] V. Vaish and A. Adams. *The (new) stanford light field archive*. `http://lightfield.stanford.edu/`. Computer Graphics Laboratory, Stanford University, 2008.

[Vag20] S. Vagharshakyan. "Densely sampled light field reconstruction". PhD thesis. Tampere University, 2020.

[VBG15] S. Vagharshakyan, R. Bregovic, and A. Gotchev. "Image based rendering technique via sparse representation in shearlet domain". In: *International Conf. on Image Processing (ICIP)*. 2015, pp. 1379–1383.

[VBG17] S. Vagharshakyan, R. Bregovic, and A. Gotchev. "Accelerated shearlet-domain light field reconstruction". In: *IEEE Journal of Selected Topics in Signal Processing (JSTSP)* 11.7 (2017), pp. 1082–1091.

[VBG18] S. Vagharshakyan, R. Bregovic, and A. Gotchev. "Light field reconstruction using shearlet transform". In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* 40.1 (2018), pp. 133–147.

[VBG20] S. Vagharshakyan, R. Bregovic, and A. Gotchev. "Densely-sampled light field reconstruction". In: *Real VR – Immersive Digital Reality: How to Import the Real World into Head-Mounted Immersive Displays*. Springer International Publishing, 2020, pp. 67–95.

[VRA+07] A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin. "Dappled photography: mask enhanced cameras for heterodyned light fields and coded aperture refocusing". In: *ACM Tran. on Graphics (TOG)* 26.3 (2007), 69:1–69:12.

[VSB+18] S. Vagharshakyan, O. Suominen, R. Bregovic, and A. Gotchev. *ICME 2018 grand challenge on densely sampled light field reconstruction*. https://civit.fi/icme-2018-grand-challenge/. 2018.

[VWJ+04] V. Vaish, B. Wilburn, N. Joshi, and M. Levoy. "Using plane+ parallax for calibrating dense camera arrays". In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. Vol. 1. 2004, pp. I–I.

[WJV+05] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. "High performance imaging using large camera arrays". In: *ACM Tran. on Graphics (TOG)*. Vol. 24. 3. 2005, pp. 765–776.

[WLW+18] Y. Wang, F. Liu, Z. Wang, G. Hou, Z. Sun, and T. Tan. "End-to-end view synthesis for light field imaging with pseudo 4DCNN". In: *European Conf. on Computer Vision (ECCV)*. 2018, pp. 340–355.

[WMJ+17] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu. "Light field image processing: an overview". In: *IEEE Journal of Selected Topics in Signal Processing (JSTSP)* 11.7 (2017), pp. 926–954.

[WOO+19] H. Watanabe, N. Okaichi, T. Omura, M. Kano, H. Sasaki, and M. Kawakita. "Aktina vision: full-parallax three-dimensional display with 100 million light rays". In: *Scientific reports* 9.1 (2019), pp. 1–9.

[WS16] O. Wasenmüller and D. Stricker. “Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision”. In: *Asian Conf. on Computer Vision (ACCV) Workshops*. 2016, pp. 34–45.

[WZW+17] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu. “Light field reconstruction using deep convolutional network on EPI”. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 1638–1646.

[XCW+19] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman. “Video enhancement with task-oriented flow”. In: *International Journal of Computer Vision (IJCV)* 127.8 (2019), pp. 1106–1125.

[XLS+19] X. Xu, S. Li, W. Sun, Q. Yin, and M.H. Yang. “Quadratic video interpolation”. In: *Conf. on Neural Information Processing Systems (NeurIPS)*. 2019, pp. 1647–1656.

[XMN+15] Y. Xu, K. Maeno, H. Nagahara, and R.-I. Taniguchi. “Camera array calibration for light field acquisition”. In: *Frontiers of Computer Science* 9.5 (2015), pp. 691–702.

[Yam16] M. Yamaguchi. “Light-field and holographic three-dimensional displays”. In: *Journal of the Optical Society of America A* 33.12 (2016), pp. 2348–2364.

[YEB+02] J. C. Yang, M. Everett, C. Buehler, and L. McMillan. “A real-time distributed light field camera.” In: *Eurographics Workshop on Rendering (EGWR)*. 2002, pp. 77–86.

[YHC+18] H. W. F. Yeung, J. Hou, J. Chen, Y.Y. Chung, and X. Chen. “Fast light field reconstruction with deep coarse-to-fine modelling of spatial-angular clues”. In: *European Conf. on Computer Vision (ECCV)*. 2018, pp. 137–152.

[Yu17] J. Yu. “A light-field journey to virtual reality”. In: *IEEE MultiMedia* 24.2 (2017), pp. 104–112.

[YZD+15] L. Yang, L. Zhang, H. Dong, A. Alelaiwi, and A. El Saddik. “Evaluating and improving the depth accuracy of Kinect for Windows v2”. In: *IEEE Sensors Journal* 15.8 (2015), pp. 4275–4285.

[Zac14] C. Zach. “Robust bundle adjustment revisited”. In: *European Conf. on Computer Vision (ECCV)*. 2014, pp. 772–787.

[ZC04] C. Zhang and T. Chen. “A self-reconfigurable camera array”. In: *Eurographics Symposium on Rendering (EGSR)*. 2004, pp. 243–254.

[ZDW10] S. Zinger, L. Do, and PHN de With. “Free-viewpoint depth image based rendering”. In: *Journal of Visual Communication and Image Representation* 21.5 (2010), pp. 533–541.

[Zha00] Z. Zhang. “A flexible new technique for camera calibration”. In: *IEEE Tran. on Pattern Analysis and Machine Intelligence (TPAMI)* 22.11 (2000), pp. 1330–1334.

[ZMD+16] P. Zanuttigh, G. Marin, C. Dal Mutto, F. Dominio, L. Minto, and G. M. Cortelazzo. *Time-of-flight and structured light depth cameras: technology and applications*. Springer International Publishing, 2016.

[ZSN+17] A. Zeng, S. Song, M. Nießner, M. Fisher, J. Xiao, and T. Funkhouser. “3DMatch: learning local geometric descriptors from RGB-D reconstructions”. In: *Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 199–208.

[ZTF+18] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely. “Stereo magnification: learning view synthesis using multiplane images”. In: *ACM Tran. on Graphics (TOG)* 37.4 (2018), 65:1–65:12.

[ZVK+17] M. Ziegler, R. op het Veld, J. Keinert, and F. Zilly. “Acquisition system for dense lightfield of large scenes”. In: *3DTV-CON*. 2017, pp. 1–4.

[ZZY+13] C. Zhu, Y. Zhao, L. Yu, and M. Tanimoto. *3D-TV system with depth-image-based rendering*. Springer Science+Business Media, 2013.