Maritime Augmented Reality mit a priori Wissen aus Seekarten

Maritime Augmented Reality with a priori knowledge of sea charts

Dissertation

zur Erlangung des akademischen Grades Doktor der Ingenieurwissenschaften (Dr. - Ing.)

der Technischen Fakultät
der Christian-Albrechts-Universität zu Kiel

Dipl. - Inf. Kristine Haase

Kiel

2012

1. Gutachter:
Professor Dr. - Ing. Reinhard Koch,
Christian-Albrechts-Universität zu Kiel

2. Gutachter:
Professor Dr. Stefan Müller ,
Universität Koblenz · Landau

Datum der mündlichen Prüfung: 13. 09. 2012

Maritime Augmented Reality with a priori knowledge of sea charts

Keywords: Augmented Reality, Topology S-57, ENC, Object Detection, Bayesian Network

Abstract

The main objective of this thesis is to provide a concept to augment maritime sea chart information into the camera view of the user. The benefit is the simpler navigation due to the offered 3D information and the overlay onto the real 3D environment.
In the maritime context special conditions hold. The sensor technologies have to be reliable in the environment of a ship's ferrous construction. The augmentation of the objects has to be very precise due to the far distances of observable objects on the sea surface. Furthermore, the approach has to be reliable due to the wide range of light conditions. For a practical solution, the system has to be mobile, light-weight and with a real-time performance. To achieve this goal, the requirements are set, the possible measurement units and the data base structure are presented.
First, the requirements are analyzed and a suitable system is designed. By the combination of proper sensor techniques, the local position and orientation of the user can be estimated. To verify the concept, several prototypes with exchangeable units have been evaluated. This first concept is based on a marker-based approach which leads to some drawbacks.
To overcome the drawbacks, the second aspect is the improvement of the system and the analysis of markerless approaches. One possible strategy will be presented. The approach uses the statistical technique of Bayesian networks to vote for single objects in the environment. By this procedure it will be shown, that due to the a priori information the underlying sea chart system has the most benefit. The analysis of the markerless approach shows, that the sea charts structure has to be adapted to the new requirements of interactive 3D augmentation scenes. After the analysis of the chart data concept, an approach for the optimization of the charts by building up an object-to-object topology within the charts data and the Bayesian object detection approach is presented. Finally, several evaluations show the performance of the implemented evaluation application.

Acknowledgments

I would like to thank Professor Reinhard Koch and the MIP working group for giving support, discussion approaches and a systematical view to the topics. Especially Arne, thanks for the most detailed discussions and the searching for black holes in my source code.
Next, I thank the Universities of Kiel and Koblenz for providing well-founded educations, the environment to learn and the equipment to research. For the continued support of the work by the supply of ECDIS data, I am very thankful to the SevenCs GmbH. And I thank Chris to walk the way with me.
Last but not least, my many thanks go to my parents, who bought the Robotron KC 87 for their children, when I was seven years old. They did not know, but this work is the long-term consequence.

Contents

List of Figures

List of abbreviations

ADE	Application Domain Extension
AI	Artificial Intelligence
AIS	Automatic Identification System
AR	Augmented Reality
ARS	Augmented Reality System
BN	Bayesian Networks
CCS	Camera Coordinate System
CDF	Cumulative Distribution Function
CLT	Central Limit Theorem
DAG	Directed Acyclic Graph
ECDIS	Electronic Chart Display and Information System
EKF	Extended Kalman Filter
ENC	Electronic Navigational Charts
FOV	Field of View
GIS	Geographic Information System
GMM	Gaussian Mixture Model
GPS	Global Positioning System
HMD	Head Mounted Displays
HMM	Hidden Markov Model
HUD	Head-Up-Displays
IALA	International Association of Lighthouse Authorities
ICA	Independent Component Analysis
ICS	Image Coordinate System
IHO	International Hydrographic Organization
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
MAP	Max A Posteriori Estimation
MF	Markov Fields
ML	Machine Learning
MN	Markov nets
MPE	Most Probable Explanation
MR	Mixed Reality
MRF	Markov Random Fields
NMEA	National Marine Electronics Association
NN	Neural Nets
OST	Optical-See-Through
PCA	Principal Component Analysis
PCS	Pixel Coordinate System
PDF	Probability Density Function
PR	Pattern Recognition
PTU	Pan-Tilt Unit
RMS	Root Mean Square Error

SDK ..... Software Development Kit
SLAM ..... Simultaneous Localization and Mapping
SVM ..... Support Vector Machines
VR ..... Virtual Reality
VST ..... Video-See-Through
WCS ..... World Coordinate System

List of Notation

$\tilde{\mathbf{p}}$	Homogeneous vector
x,y,z	Scalar values
p^T	Transposed point
M	Matrix
$\mathbb{R}^n$	Mathematical set
$\mathcal{S}$	Subset of a mathematical set
$\mathcal{N}$	Distribution
$\mathbb{R}^n \rightarrow \mathbb{P}^n$	Transformation
$\mathcal{G}$	Graph
$\mathbf{x}$	Vector
$p^{'}$	Mapped value
$\times$	Cross product
x	Cartesian product
$\longmapsto, -, \rightleftharpoons$	Direction between nodes in graphs
$\cup, \cap, \subset, \supset$	Set algebra
$prob$	Probability
X,Y	Random variables
P	Stochastic process
$\mathbb{V}$	Variance
$\mathbb{E}$	Expectation

Chapter 1

Introduction

Contents

1.1 What is Augmented Reality? Idea and Vision

A possible description of *Augmented Reality* (AR) is the following: Every context driven display of data in an interactive, real-time representation of the actual environment is an augmentation. It is an interface between the real world and the knowledge of a virtual data base. The challenge in this technology is the proper positioning of the enhanced data. An example of an AR application could be the enhancement of a car component for a technician with data from physical simulations.
The interest in AR is increasing in recent years. Nowadays many applications exist that use the base technology of AR, spreading the topic of AR to many users. In most cases the quality of these tools is not sufficient enough to obtain a reliable application in real circumstances. The main contribution of these tools is entertainment and to increase the sensitivity for this technology in the young generation. To develop a robust and reliable software, an *Augmented Reality System* (ARS) has to solve the following issues accurately and in real-time:

- The capturing of the real world.
- The estimation of the actual pose of the user/AR System in 6 *Degrees of Freedom* (DOF).
- The access and generation of virtual data.
- The correct positioning of the virtual data in the line of sight.
- The handling of user interactions.
- Manageability and spatial mobility.

Nowadays, AR mainly stands for a vision of an exhaustive exploitation of the technical improvements in the civil domain. The modern knowledge based society has expanded its knowledge into the virtual space and expects the usage of all available information at any given time. Since a single human has not an all-encompassing knowledge anymore, and the data in the everyday life grows rapidly, AR stands for a practical solution for an efficient data retrieval and display. General knowledge that is always available due to the World Wide Web, does not need to be learned anymore. This holds for instruction manuals, terminologies and chart data (environmental descriptions). The available data can also enhance the human senses in form of measurements, e.g. of temperature or wind sensors. This leads to the possibility of an improved decision-making due to an increased knowledge base. Due to the networking aspect of the virtual database, information is available anytime and anywhere. To sum up, the vision is to obtain an additional benefit for the actual situation by using a general knowledge base without effort. To be able to do this, technologies are necessary that measure the position accurately and understand the context of the situation by the help of a given data base. The next step is to prepare the amounts of data for an efficient usage.

1.2 Topic: Augmented Reality and 3D Sea Charts

Few researcher address the problem of safe navigation with the help of new technologies. A pioneer in this topic is Porathe [156, 155], who explains the problem in the following way: Shipping accidents occur, even if daily routes are driven that are well known by the nautical staff. Commonly, the explanation is the "human error". The problem of ship navigation is given by very many sources of information that are given in different ways. The classical charts are 2D projections with known limitations like the equality of angles or areas and given in a top-down perspective. The orientation is varying between north-up for paper charts and adapted head-up orientation for electronic charts. In contrast to them, the real view of a person is an interpreted 2D image on the retina with originally upside-down orientation.
Beside them, there exists a mental map drawn by the own experience of a route. To compare all of these maps, the nautical staff has to find correspondences like angles or landmarks and has to fulfill mental map rotations. In experiments Porathe showed, that this task can not be learned in a way that nautical staff can be faster than inexperienced users [160]. But in the same experiments he pointed out, that errors are reduced if nautical staff and native users try to navigate with 3D charts. Errors in navigation or short-term confusions can be corrected by optical comparisons. If the weather conditions are bad like fog, rain or darkness, this aid fails and the risk of accidents increases dramatically.
The chance of 3D charts to minimize the risk of errors in the ship navigation is an actual topic and opens the view to many research topics.
3D charts have to be developed and efficient display possibilities have to be reviewed. At the end, an expedient integration on the ship's bridge and into the working processes has to be evaluated.

1.3 System Requirements

Goal of this work is the analysis of the possibilities of ARS for the maritime usage. In this context, special requirements hold. Normally, the nautical staff on the ship's bridge uses a binocular and the knowledge of their own position in sea charts to observe navigational aids. A binocular is a magnifying device which is used in free hands mode anywhere on the ship without the preparation of the environment. The free hands operation causes fast local rotations in three dimensions independent from the ships movements. In contrast, the user movements are dependent to the location where the ship's position is measured, usually a GPS-antenna.
The objects to observe can be looked up in the actual sea chart and range at distances between 100 m up to approximately 10 km. As magnification typically a factor 6-7 with a field of view of 7x10 degree is used, which leads to directional errors that have a big effect on the augmentation. Using a camera with similar characteristics, namely a resolution of 640x480 pixel with a 25 mm objective on a 1/3 inch sensor, produces a mapping of 60 pixel per degree of the environment. To reach a precise superposition up to pixel accuracy in the augmentation, the accuracy of the estimation of the rotation angles has to be better than 0.016 degree. The effect of errors by distant objects is illustrated in [121]. Unfortunately, magnetic compass measurements will be very erroneous due to the metallic hull of the ship. To reach this accuracy for an outdoor augmentation device, an effective combination of sensors for the estimation of position and orientation has to be defined. The sea charts database can be used to get the absolute position of the observed objects in world coordinates. The ship position and orientation is known from available GPS and compass sensors on the ship. Since the relative distance between ship and object is large, the positional error due to position uncertainty of the binocular on the ship's bridge can be neglected. However, the viewing direction is needed with great accuracy.
Any application which has to deal with these requirements has to reach real-time performance to be able to measure the fast rotations of the user and to minimize delay errors between the components of the hand-held device. The aim is to define an augmentation device that can be used like a conventional binocular on the ship's bridge. Due to the error-prone measurements in the hardware, a refinement of the pose estimation is developed. It will be shown, that an object detection approach which is based on the a priori information of the given sea charts can significantly support the functionality of such devices.

1.4 Thesis Focus

The figure 1.1 shall serve as an overview of the augmentation process. To develop such a maritime augmentation system, the single steps in the data flow have to be analyzed and evaluated. At first there will be the derivation of a fundamental concept that follows the given requirements. For this purpose, the estimation of the pose has to be evaluated and a fitting sensor system has to be chosen. This basic system has be evaluated to mark further improvement possibilities.
The basic concept will be realized by a marker-based approach to verify the sensor

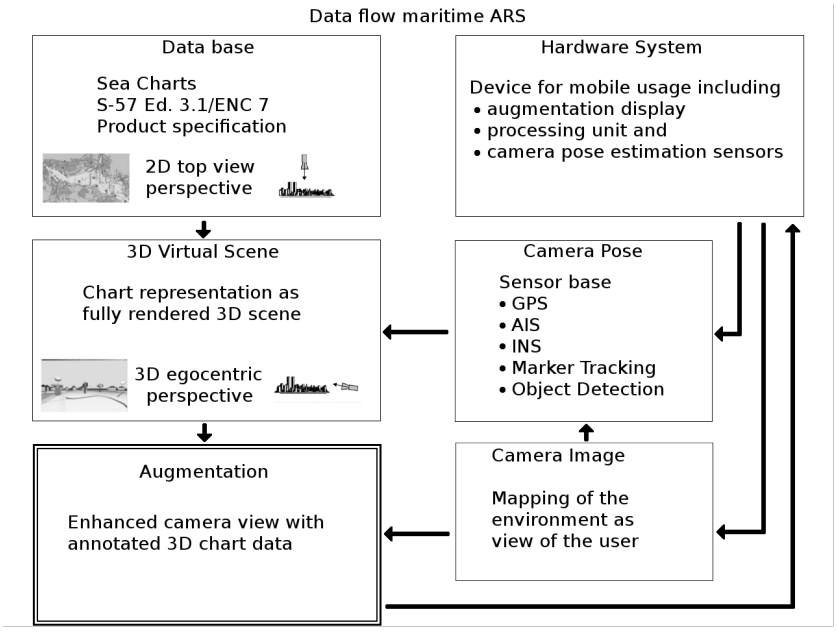

Figure 1.1: Overview of a maritime Augmented Reality System

fusion concept. Furthermore, it will be shown, that a markerless approach will improve the movement range of the user and give the chance to use the system outside the bridge, which is very interesting for nautical staff.
To be able to build an information model by the sea charts to support the markerless approach, the sea charts have to be analyzed and restructured in their object-to-object topology.
To manage unprepared environments with a markerless approach, the thesis will focus on an a priori knowledge model that can be referred to the underlying sea chart system. The inference model will be defined with an underlying Bayes framework and the representation in Bayesian networks.
To sum up the main topics, this work will focus on:

1.4.1 The Development of an AR System for Maritime Application

The goal is the development and evaluation of a new concept of a device for maritime augmented reality. The device has to fulfill the mentioned requirements, that refer to hardware and software skills. Another goal is a light-weight hardware that estimates in real-time the precise pose for the augmentation. The concept should be modular and with exchangeable sensor types. It will be shown, that this concept has been developed and due to the actuality of this topic, the device has been patented.

1.4.2 The Analysis of 3D Sea Charts for AR Applications

The developed Augmented Reality System has been evaluated with a marker-based image processing approach that restricts the usage to an indoor environment. To

increase the range of the user's movements, a markerless approach is provided. This new approach uses the sea charts as a priori information. To obtain usable data from the sea charts, the charts have to be optimized for this context. Due to this, one basic goal is the analysis of the standardized sea charts to find a starting point for the adaption.

1.4.3 The Optimization of 3D Sea Charts for AR Applications by Topology

The sea charts do not provide an object-to-object topology that it would be very useful to support an interactive scene management for the users and to guide the user in the structuring of augmentation scenes. The analyzing of the sea charts lead to a possibility to build a topology within the sea charts objects. These relations can be stored within the standard. Furthermore, there can semantical relations be built in parallel. Unfortunately, these newly generated data can not be stored in the format. As alternative, a suitable exchange format will be suggested and explained.

1.4.4 The Development of an Approach with Bayesian Networks of 3D Sea Charts for Precise Pose Estimation

The core of the markerless approach is the usage of a priori information to solve the pose estimation problem. The belief of what could be observed in an image can be used to identify single objects. With this procedure, the user will be able to continuously recalibrate the orientation estimation method and to overcome drift effects of the sensors. The statistical data will be collected into Bayesian networks and inferenced to get not only a classification of the object type, but also the single instance of this class.
Partial results which are presented here have been published beforehand in these publications:

- [72] Haase, K. and Koch, R., 2008. Augmented Reality System for nautical navigation, Mobile and Embedded Interactive Systems, Lecture Notes in Informatics Series, GI, ISBN 978-3-88579-227-7
- [74] Haase, K. and Koch, R., 2010. Extension of Electronical Nautical Charts for 3D interactive Visualization via CityGML, Proceesings of GeoInformatik, Heidelberg, ISBN 978-3-89838-335-6
- [75] Haase, K. and Koch, R., 2010. Extension of Sea Charts for 3D Visualization, ISPRS proceedings, 3DGeoInfo, Volume XXXVIII-4/W15, ISSN 1682-1750, Page(s) 129-134
- [73] Patent DE102008023439B4, February, 17th 2011, K. Haase, R. Koch, Augmented Reality Fernglas zur Navigationsunterstützung

Chapter 2

Theoretical Foundations

Contents

2.1 Mathematical Basics

2.1.1 Sensor Technologies

The core of an efficient ARS is the pose estimation and its characteristics. Multiple sensors can be combined to achieve optimal results for the actual scope. This section describes the different coordinate systems, the camera models and the translation and rotation sensors. In advance, as an introduction to the imaging process and basic characteristics, see the appendix A.1.2.

2.1.1.1 Camera Models and Projective Geometry

The mapping of the 3D *World Coordinate System* (WCS) onto a 2D *Image Coordinate System* (ICS) underlies the mathematical foundations of projections. The simplest camera model imaginable is the *central projection* that is given in the pinhole camera, visualized in figure 2.1. A coordinate system $C\,(X, Y, Z)$ is attached to the camera center (*Camera Coordinate System* CCS), called pinhole. The plane spanned by the axes X and Y lies parallel to the image plane, with center P lying on the positive Z axis as the origin of the ICS $P\,(x, y, z)$. The Z axis can be interpreted as vector, crossing the camera center and the image plane. This forms the *principal axis* of a camera. The distance between the center of C and P is the *focal length* f and it is the base for the magnification of the resulting mapping, as a point $X = (X, Y, Z)$ will be mapped by

$$(X, Y, Z)^T = (fX/Z, fY/Z, f)^T \tag{2.1}$$

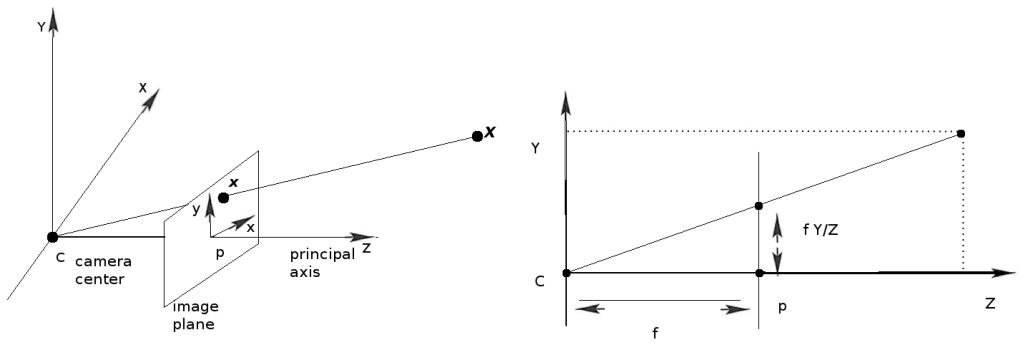

Figure 2.1: Pinhole camera geometry, based on [76].

if $z = f$. Ignoring the third coordinate, results in the mapping from $\mathbb{R}^3$(WCS) to $\mathbb{R}^2$(ICS). On the image plane, the *Pixel Coordinate System* (PCS) can be defined. The origin lies in the upper left corner of the image and therefore differs from the image-centered principal point P. The *central perspective* projection is a simple approximation of the mapping process. Other projection models are *orthographic* projection and the *weak perspective* projection. The orthographic projection maps a scene point $X = (X, Y, Z)$ to an image point x $= (x, y)$ by simply ignoring the depth coordinate. The weak perspective projection combines the other two by projecting orthographically at first and perspectively afterwards. Normally, the resulting images are distorted, due to the fact that the reality is not ideal. Distortions can be caused by a translated image center, a distorting lens (radial, tangential as well as aberrations, skew and vignetting effects), not perfectly squared pixel elements on the sensor chip (aspect ratio) or if the principal axis is not perpendicular to the projection plane. The internal characteristics of a camera can be summed up in the so-called *K-matrix*

$$K = \begin{pmatrix} f_x m_x & s & x_0 \\ 0 & f_y m_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.2}$$

with $f_x m_x$, $f_y m_y$ as focal length in terms of pixel dimensions in x- and y-direction, s as skew parameter of the sensor and x_0, y_0 as coordinates of the principle point. To reconstruct the ideal conditions as a base for geometrical estimations, a calibration process calculates the parameters. In the reality the coordinate systems are not as perfectly aligned as presented. To transform a point within a coordinate system, we use a translation vector $\mathbf{t}$ and an affine transformation, which can be a scaling, a shearing or a rotation. These transformations can be combined, but they are not commutative. A rotation matrix R with *Euler* angles Θ, Ψ, Φ with X, Y, Z order is defined by

$$\begin{pmatrix} \cos\Psi\cos\Phi & -\cos\Theta\sin\Phi + \sin\Theta\sin\Psi\cos\Phi & \sin\Theta\sin\Phi + \cos\Theta\sin\Psi\cos\Phi \\ \cos\Psi\sin\Phi & +\cos\Theta\cos\Phi + \sin\Psi\sin\Theta\sin\Phi & -\sin\Theta\cos\Phi + \cos\Theta\sin\Psi\sin\Phi \\ -\sin\Psi & \sin\Theta\cos\Psi & \cos\Theta\cos\Psi \end{pmatrix}. \tag{2.3}$$

Rotation matrices have special characteristics and form a linear group in the Euclidean space as a subset of orthogonal matrices. The exact definitions are widely known and can be found in the appendix A.1.4 or in [76], especially in combination with point and coordinate transformations. To perform a transformation of this kind, a non-linear estimation is necessary, as it has to deal with a vector addition and a matrix multiplication. To linearize this for computational estimations there exists the mathematical solution of using a higher dimensional space. By expanding the $\mathbb{R}^n$ by one dimension, we get the *projective space* $\mathbb{P}^n$. Now, not only rotations and translations, but also the projections, usually defined by an affine group of transformations can be modeled likewise. The formal representation of $\mathbb{P}^n$ is given by

$$\mathbb{R}^n \rightarrow \mathbb{P}^n : \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \rightarrow \alpha \begin{pmatrix} wx_1 \\ \vdots \\ wx_n \\ w \end{pmatrix}, \text{ with } \mathbb{P}^n = \left\{ x \in \mathbb{R}^{n+1} \right\} \tag{2.4}$$

for $x \neq 0$ and $\alpha \in \mathbb{R}/\{0\}$. For the projective space we use the notation of homogeneous coordinates, here written as $\tilde{\mathbf{p}}$. Points in the projective space are equal, if they span a subspace in the vector space. As a result, points in the projective space are only defined up to a scaling. This also holds for transformations. Correspondingly, points would be defined at infinity in the *Euclidean space*. *Vanishing points* can be modeled this way. One benefit of the projective space is, that we can define a 2D-translation and a 2D-rotation in $\mathbb{P}^2$ by a single matrix H,

$$H = \begin{pmatrix} A & \mathbf{t} \\ \mathbf{v}^T & \mathbf{s} \end{pmatrix} \tag{2.5}$$

with A as affine transformation and the vanishing line vector $\mathbf{v}^T = (v_1, v_2)$, resulting in $\tilde{\mathbf{p}}' = H\tilde{\mathbf{p}}$ for transforming a single point with eight DOFs. The projective geometry has several characteristics like

- four points are sufficient to model a projective transformation
- transformation groups by chains of single transformations can be built
- all points of an equivalence class are mapped to the same point $\tilde{\mathbf{p}}$ in $\mathbb{P}$
- by defining the ideal points with a finite value, it can model vanishing points
- invariant properties are concurrency, collinearity, intersection and cross ratio

Looking at the hierarchy of euclidean, similarity, affine and projective geometry, the projective geometry is the most general class. Transformations like rotation, translation, isotropic scaling, shearing and perspective and collinear projections are modeled whereas the only invariants are the cross-ratio and the incidence. Between two images (concerning 3D objects, image planes and camera centers) exist bilinear constraints and geometric dependencies (*epipolar geometry*) [76] [236] which can be found in the formulation of the *fundamental matrix* (F-matrix). The F-matrix is classically estimated by algorithms that exploit the singularity constraints in combination with a robust estimator like *RANSAC*. Between 2 or more F-matrices trifocal constraints can be given and in combination with intensity values the trifocal tensor can help to estimate motion estimations. If the camera centers are identical,

the *infinity homography* H_∞ can be used to estimate the correspondence between points. In this case, H describes the mapping to Π_∞ by $H_\infty \cong P_i P_j^{-1}$, where P_i and P_j are projection matrices. The well-defined mathematical background of the projective space will not be covered here in detail, further details can therefore be found in [63, 76]. The overall perspective mapping becomes

$$\widetilde{\mathbf{p}}_{PCS} = K * H * D * \widetilde{\mathbf{p}}_{WCS} = P\widetilde{\mathbf{p}}_{WCS} \tag{2.6}$$

with D as the matrix of extrinsic parameters and K as matrix the of intrinsic parameters. Note that the mapping includes the coordinate transformations from WCS → CCS → ICS → PCS [146][65][17][39].

2.1.1.2 Position Estimation Techniques

For an introduction to the used coordinate systems, see the appendix A.1.3. It is essential for an ARS to estimate the most accurate position and orientation. All augmentations in the video frame depend on these points in their overlay precision. Imagine a video frame of 640x480 pixels, with a *Field of View* (FOV) of approximately 10 degrees in horizontal range. This is comparable to a binocular and implies that 1 pixel covers 0.016 degree. An augmentation with a pixel error of 50 pixels is very large and well observable as it is 1/12 of the whole view but it only refers to 0.78 degree. That shows how important the overall accuracy in position and orientation estimation is, due to an error-free augmentation. Several positioning and orientation measurement methods are available and this section presents the ones important for this work.

2.1.1.2.1 GPS The *Global Positioning System* (GPS) is a satellite based navigation system. The actual system is the Navigation System for Timing and Ranging (NAVSTAR), maintaining 24 satellites. The specification is given in [70]. The concept is to listen to satellite signals with a local receiver on earth, compute the time difference ($\Delta\nu$) and to estimate the distance to the satellite. For this estimation the satellites time stamp has to be submitted by the signal and the speed of the signal has to be known, which is constantly the speed of light (c):

$$distance = \Delta\nu * c. \tag{2.7}$$

By having the distance of one satellite, the position can be narrowed down to a circle, see figure 2.2 (a). Only by having at least four satellites the values for position, height and time error can be estimated [239]. See figures 2.2 (b) and (c) for an illustration of two and three measurements [228].
Having ideal measurements, the trilateration of the distances of the three satellites would yield a single point as localization. Due to many error sources, this does not hold in reality and four measurements are taken. The so-called *pseudo range* is shown in figure 2.2 on the right hand sketching an area of interest as pseudo range $= r + \Delta\varepsilon * c$, with r as correct distance and $\Delta\varepsilon$ as time error between satellite and receiver.
Receivers have to deal with the pseudo range by adequate algorithms that can solve

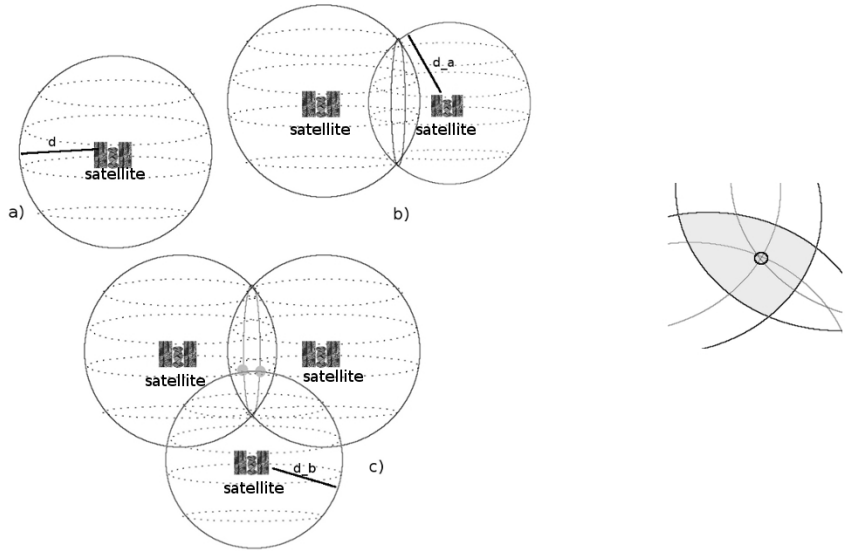

Figure 2.2: Left: Illustration of the position estimation dependent on the number of satellites with a)one satellite, b)two satellites and c)three satellites. Right: The pseudo range is the area between the circle lines, based on [114].

non-linear equations. The time of the signal is calculated by correlation time shifts. The GPS signal delivers at least information about satellite time-of-transmission, satellite position, satellite health, constellation status, clock correction, ionospheric delay effects and the time transfer to *Coordinated Universal Time* (UTC) as kept by the U.S. Naval Observatory. Error sources for the estimations are widespread and can affect the satellite, the receiver or the signal itself.
To mention the most affecting: clock errors (time has to be accurate up to 10 nanoseconds), multi-path effects caused by reflections, ionospheric effects, shifts in the satellite orbit, tropospheric effects, quality of satellite geometry (Dilution of Precision DOP) and of course noise and rounding errors. See figure 2.3 for an overview of the importance of the errors, given by [197]. A GPS receiver can provide position, height and velocity information about itself. Various errors can be minimized by adding GPS signal stations on earth (Differential GPS). These stations can provide precise correction information as they include local constants. The *World Geodetic System 1984* (WGS84) is commonly used as reference coordinate system by GPS sensors.
To transform the spherical coordinate (latitude φ, longitude λ) into Cartesian coordinates, these equations hold

Error cause	without DGPS	with DGPS
Ephemeris data	1.5	0.1
Satellit clocks	1.5	0.1
Ionosphere effect	3.0	0.2
Troposphere effect	0.7	0.2
Multipath reception	1.0	1.4
Reciever effect	0.5	0.5
RMS value, total	4.0	1.2
Error horizontal (1-sigma(68%) HDOP=1.3)	6.0	1.8
Error horizontal (2-sigma(95%) HDOP=1.3)	12.0	3.6

Figure 2.3: Overview of error types and their importance in meters, on the base of [239].

$$\mathrm{N}_\varphi = \frac{a}{\sqrt{1 - e^2 * \sin(\varphi)^2}} \text{ with} \tag{2.8}$$

$$x = (\mathrm{N}_\varphi - h) * \cos(\varphi) * \cos(\lambda) \tag{2.9}$$

$$y = (\mathrm{N}_\varphi - h) * \cos(\varphi) * \sin(\lambda) \tag{2.10}$$

$$z = \left(\mathrm{N}_\varphi * 1 - e^2 - h\right) * \sin(\varphi), \tag{2.11}$$

with a as the semi-major axis of the earth ellipsoid, e as eccentricity and h as height.

2.1.1.2.2 AIS The *Automatic Identification System* (AIS) is a radio communication system (very high frequency - VHF) for the shipping traffic with the goal to preserve accidents and collisions of ships. In the specifications [92], an AIS message is defined by the information of the ship's static data (IMO and MMSI number, name, ship's type, ship's dimensions), dynamic data (navigation status, position, course over ground, heading, rate of turn) and travel data (hazardous materials, destination and estimated time of arrival). According to [91] the AIS has to be installed on professional ships but in contrast to them, anyone with an AIS receiver can have an overview of other ships in the closer range. The AIS is important for this work, because the system used on a ship can therefore deliver the ship's data by reading the AIS-based pilot plug. AIS messages are conform with the specifications of the *National Marine Electronics Association* NMEA[1]. The NMEA protocol is a standard for data transfer in the field of maritime and GPS and is used to unify the different encodings by a common data format. Every machine or sensor on the ship has to push their data on the ship's NMEA bus in this standard. Different standards have been developed like the NMEA0183, NMEA0400 and the NMEA2000. The NMEA2000 is the newest, whereas NMEA0183 is still the most used. For consistence, the AIS messages are conform and the signals are sent with 38,400 baud and each message has to start with the string *!AIVDM* or *!AIVDO* as identifier. In contrast to standard NMEA messages which are in plain text like "$GPGGA, time, latitude, south or north of the Equator, longitude, east or west of zero meridian, quality, number of satellites, horizontal dilution of precision, height over geoid, unit,

[1] http://www.nmea.org/

height (geoid minus ellipsoid), unit, checksum.", the AIS sentences include binary data. By the definition in ITU 1371-1 [92] there are 22 of 64 possible message types defined. An overall message looks like "!AIVDM, message part, sentence size, message ID, channel code, data bit vector, checksum". For the purpose of this work up to now it is satisfactory to decode message 21, which holds the position and status report [93]. Note that there is an extra standard for the usage of inland waterways [238].

2.1.1.3 Rotation Estimation Techniques

2.1.1.3.1 IMU Modern *Inertial Measurement Units* (IMU) consist of digital accelerometers, gyroscopes and often additional compasses. Accelerometers measure their inertial acceleration along their specific axis. By integration of the output, the estimation of velocity and position can be done. Three accelerators and therefore three axes are used to cover three DOFs. As the acceleration is always relative, a starting position has to be known. Gyroscopes are also single-axis measurement devices and return the angle of motion. The gyroscopes can support the accelerometers by giving their motions the correct direction. The *Schuler tuning* is the correction of the plane of the gyroscopes, as errors occur when moving along the earth surface. If no position is given, e.g. by a GPS sensor, an IMU is a dead reckoning navigation

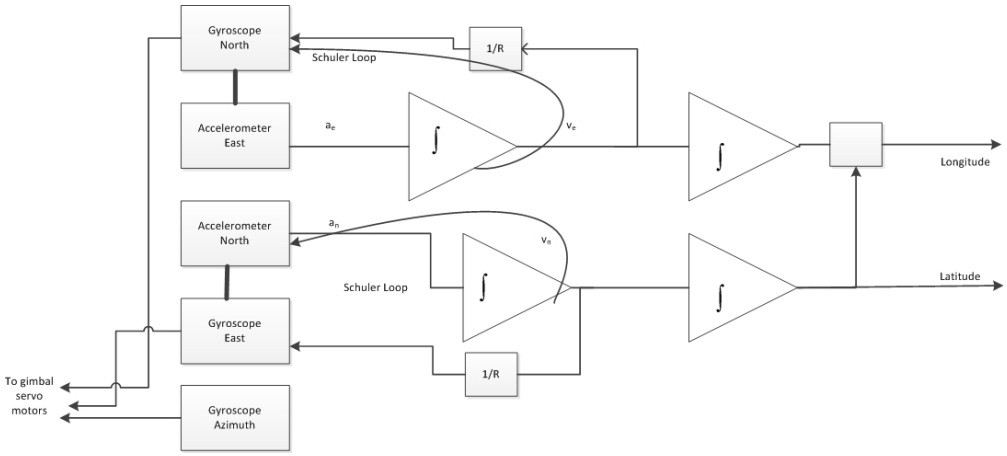

Figure 2.4: Simplified block diagram of an analogue INS on the base of [106].

system and errors will occur proportional to time and distance. Due to that, IMUs are up to now always combined with velocity or attitude updates to minimize the drift errors. In figure 2.4 an overview over a classical analogue *Inertial Navigation System* (INS) is given. INSs are highly complex systems, which allow the user to estimate its position, orientation, and velocity without external references. INSs include IMUs and improve them by providing a position estimation. They can reach accuracies of 0.01 degree standard deviation per hour. Normally these systems are very expensive and heavy of weight so that they do not fit to the requirements of a hand-held system. Therefore, IMUs are the alternative systems which are much less accurate and need an additional global reference. The most common technology

today is the one of MEMS, see [208] for a very good overview. Modern INS weight about 10 kg and are very expensive, but have a drift of 0.6 nm /hour [2]with inertial measurements only, with accuracies for attitude of 1 mrad and velocity of 0.7 m/s. IMUs have biases for attitude of 1 deg/s and velocity of 0.02 m/s^2[3]. For practical AR applications, only IMUs can be used. By modeling the geometric constraints of an INS, IMUs are bound in a mathematical strap down system, which is illustrated in figure 2.5. The IMU is able to measure the gravity, as it is the acceleration in direction to the earth center. A given gravity vector can span a horizontal plane, which is always perpendicular onto the earth surface. Often a magnetic compass is included to the IMUs to deliver the attitude update. The IMU can deliver 6 DOFs for the pose estimation of an ARS.

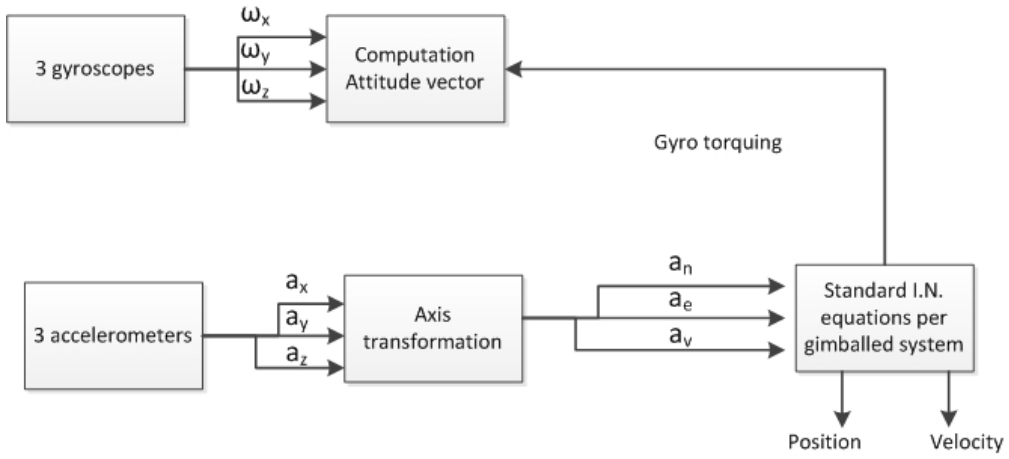

Figure 2.5: Strap down inertial unit block diagram on the base of [106].

The modeling of the error of a gyroscope is given by [228] as

$$w^b_{out'} = \mathrm{M}_{Gyro} * w^b_{out} + b_w + n_w \tag{2.12}$$

with $w^b_{out'}$ as output, M_{Gyro} for the misalignments of the gyroscopes that are ideally perpendicular to each other. b_w are systematical biases of the sensors (drift) and n_w models the white noise in the system. The same equation can be used for the error modeling of the accelerometers, but an additional centripetal acceleration has to be included if the system is heavily rotated. With the help of error propagation, the overall position error increases by

$$\Delta pos_{b_w} = \frac{1}{6} g \begin{pmatrix} b_{w,y} \\ -b_{w,z} \\ 0 \end{pmatrix} t^3 \tag{2.13}$$

with time t, b_w for all biases, $b_{\chi,y,z}$ as bias of the x,y,z component and g as acceleration [106][15][228].

[2] nautical mile $\cong 1852m$

[3] as example: http://www.xsens.com/en/general/mti

2.1.2 Graph Theory

Graph theory can help to visualize relationships in data sets and to evaluate their semantics. *Bayesian networks* (BN) 2.2.3 which will be used by this work, are based on graphical models. As base for the understanding of networks, this section will give the main definitions in the context of graph representations.
A *graph* $\mathcal{G}$ is a pair $\mathcal{G}= (\mathcal{V}, \mathcal{E})$. $\mathcal{V}$ can be interpreted as a finite set of vertices $\mathcal{V} = (\mathrm{x}_1, ..., \mathrm{x}_n)$ and $\mathcal{E} = (\mathrm{x}_i, \mathrm{x}_j) \in \mathcal{V}\mathrm{x}\mathcal{V}$ defines ordered pairs of vertices. The set of single vertices $\mathcal{V}$ is called set of *nodes* and the pairs in $\mathcal{E}$ are called *links* or *edges* between the nodes. The links can be directed $\mathrm{x}_i \longmapsto \mathrm{x}_j$ or undirected $\mathrm{x}_i - \mathrm{x}_j$. If $\mathrm{x}_i \longmapsto \mathrm{x}_j$ holds, x_j is called the *child* of x_i and vice versa x_i is called the *parent* of x_j. *Leafs* are nodes without children and the *root* is the head of the graph with no parent.
In the undirected case, the nodes are called *neighbors* and in both cases they are adjacent to each other. The *degree of a node* is defined by the number of adjacent neighbors whereas the *degree of a graph* is the maximum of the degrees of nodes. The set of parents of a node x_i is $pa(\mathrm{x}_i)$, and with the set of children $ch(\mathrm{x}_i)$, the *family* of a node is given by $fa(\mathrm{x}_i) = \{\mathrm{x}_i\} \cup pa(\mathrm{x}_i)$. The nodes are mentioned as *joined* if $\mathrm{x}_i \rightleftharpoons \mathrm{x}_j$ holds, additionally if all pairs are joined, the graph can be called *complete*. The *boundary* $bd(\mathrm{x}_i)$ is then the set neighbors and the boundary of a subset $\mathcal{A} \subset \mathcal{V}$ is defined by all vertices $\mathcal{B} \subset \mathcal{V}$ that are adjacent to $\mathcal{A}$. The *closure* is then $cl(\mathcal{A}) = \mathcal{A} \cup bd(\mathcal{A})$. A *subgraph* $\mathcal{G}_A = (\mathcal{A},\ \mathcal{E}_{\ A})$ can be built by the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, because $\mathcal{A} \subseteq \mathcal{V}$ and $\mathcal{E}_A \subseteq \mathcal{E} \cap (\mathcal{A}\mathrm{x}\mathcal{A})$. *Cliques* are complete subgraphs and are often used to describe relationships.
A *Directed Acyclic Graph* (DAG) is an often used form of a graph. An undirected graph and a DAG refer to the group of *chain graphs*. A *path* between two vertices is the sequence of nodes along the edges in a directed way, whereas a *trail* is a path between undirected vertices in $\mathcal{G}$. The next important definition is the separation characteristic. A $(\mathrm{x}_i, \mathrm{x}_j)$-*separator* is given, if the subset $\mathcal{C} \subseteq \mathcal{V}$ fulfills the requirement of intersecting all trails from x_i to x_j [110][42].
Trees are forms of graphs which are used very widely. A *polytree* is a directed graph with only single edges, in the undirected case it is called a *forest*; if there also exist trails in a directed forest (called *connected*), it is defined as tree. Important for the usage in this work are *chordal graphs* which are a method to analyze graphs. The aim is to build a *junction tree* which can be easily used by computational estimations. This subject is handled in the section 4.4.3 as an inference method in Bayesian networks. Trees are not only used to represent data, they are a classical classification approach. *Decision trees* are like a sequence of decisions. With this structure, a modeling of non-linear decision boundaries is possible. In *classification trees*, each node consists of a variable and a threshold, that decides which child is chosen for the next step. The leafs are the class labels in this context. The inference over the tree is divided into feature subsets, which are solved in a multistage process. In the case of two classes the tree is a *binary tree* and divides the feature space into two parts. If the training set contains all necessary features, trees can be pruned and show a very good generalization. The main task is to define the splitting function for the data set which decides about the structure of the tree and

the question it poses. There is a wide field of suggestions for splitting functions in the literature. The most intuitive solution is to define a quality argument, which is estimated for each possible splitting function and the searching strategy should use the minimum or maximum value. For general data sets, this leads to an extensive search. That's why many optimizing approaches can be found in literature. The other way around, pruning algorithms are designed to stop the growing of a tree before an over-fitting exists [224]. Classification trees are expandable with probability data to be more flexible. Unfortunately, a single wrong decision leads to a wrong classification. Some drawbacks are, that questions may not be answerable, a tree can not grow easily and missing data are not manageable by these rigidly structured trees [42]. The main aspect is, that they do not separate the knowledge base and the inference engine. More flexible are *causal networks*, like they are used in this work. See sections 2.2.2.2 and 4.4.3 for further information.

2.2 Machine Learning and Pattern Recognition

2.2.1 Introduction

The field of *Pattern Recognition* (PR) is embedded into the *Machine Learning* (ML) and the *Artificial Intelligence* (AI) theme. In this context, the PR is the central approach to detect objects, whereas the ML uses these information to learn concepts for the optimal process. Figure 2.6 shows the relations between the topics and the following steps for a recognition task. PR deals with the description of objects and the classification of descriptions to objects. To realize this, the field of PR has to deal with many topics like image analysis (or in general: raw data measurements), feature description, feature reduction, object (or in general: pattern) description, classification and learning approaches. The overall topic is the mapping and description of similarity. The next sections will deal with the image analysis and classification approaches referring to this work.
Here, the data measurement is an imaging process, which is a mapping of the real world onto planar sensor elements. It is attended by transformations of the mapped objects and changes in the data due to the imaging process. The task is to describe the world by the evaluation of image elements. But first, there will be an introduction to ML and PR, as it builds the basic field of this thesis.

ML approaches can be classified by different characteristics. First of all, the size of the training set is crucial. *Case-Based Reasoning* (CBR) approaches can deal with sparse data, whereas semantical or causal networks need a lot of examples. The kind of data is another attribute. *Support Vector Machines* (SVM) need empirical data. Other systems can handle logic-based grammars. The preprocessing of the data can be in a syntactical, statistical or structural manner. Syntactical approaches use grammars and symbols to describe the detection objects. Statistical applications like modern SVM or *Neural Nets* (NN) weight detection results and the data structure concepts with probabilities. The structural approaches like Bayesian Networks combine different methods. If the classification of the training data is given, there has been a *supervising*. An example is the *k-Nearest Neighbor* approach. *Unsupervised* methods can deal with unlabeled data like the *k-Means method*. The next

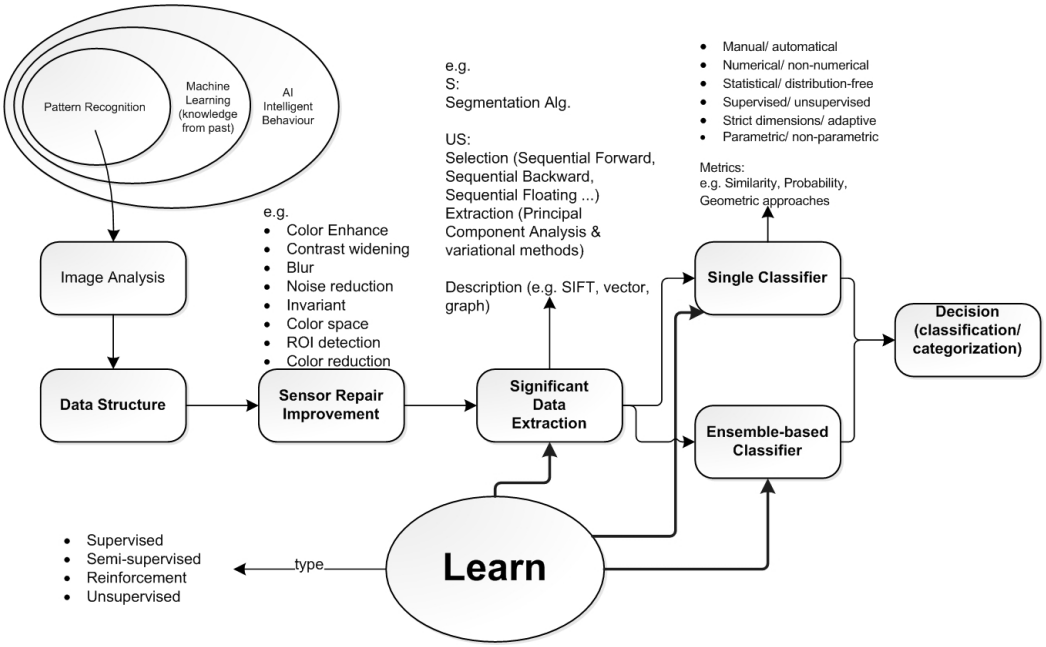

Figure 2.6: Pipeline of the thematic of the recognition process.

question is, if the learning will be *incremental* or not. An incremental learning can use the trainings data to improve the results step by step and can easily grow the trainings data [51][224][127].

2.2.2 Probability Theory

In cases where supervised learning or logic-based approaches can not deal with missing data or uncertainty and where maximum likelihood estimations fail in small data sets, the usage of initial guesses in form of a priori information can help to model a learning approach [207, 222]. Figure 2.7 provides a brief overview to the probability theory. One mathematical topic is the analysis of non-deterministic events. In this stochastic field the statistic domain deals with the observation and the results of derivations of prediction models for observed data. Based on the three subjects of statistics (descriptive, inferential, inductive), the discriminant probability theory refers to the inductive interpretation of statistics. Inductive inference describes the approach to learn general concepts by observing training data, whereas descriptive inference works with the given data without derivations. In contrast to them the inferential statistics form hypotheses by descriptive approaches and tests them by inductive experiments to verify them, mixing both previously mentioned types. Assuming that the given events are randomly distributed, the probability theory uses measured quantities to describe the measurement of outcomes, called *probability*. To describe the probability distribution of all quantities, we have to define the events and their space of outcomes, Ω. The measured data is then a subset $\mathcal{S}$, consequential each event $\alpha \in \mathcal{S}$ is a subset of Ω. A probability distribution $\mathcal{P}$ over

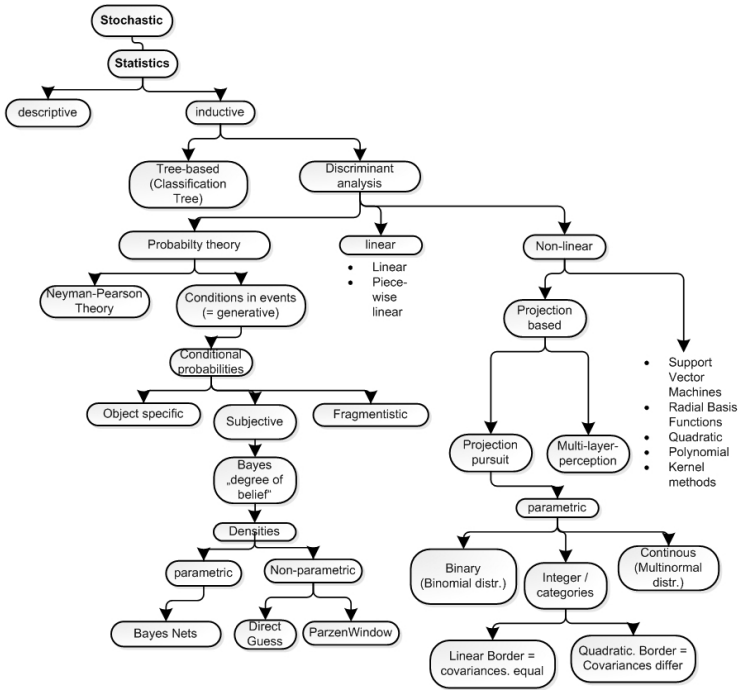

Figure 2.7: Overview of the relations of relevant topics in the field of statistic approaches.

$(\Omega, \mathcal{S})$ satisfies

- $\mathcal{P}(\alpha) \geq 0$ for all $\alpha \in \mathcal{S}$
- $\mathcal{P}(\Omega) = 1$ and
- if $\alpha, \beta \in \mathcal{S} \wedge \alpha \cap \beta = \emptyset$, then $\mathcal{P}(\alpha \cup \beta) = \mathcal{P}(\alpha) + \mathcal{P}(\beta)$.

To model the outcomes, we define a set of random variables [110], as we assume the events to be randomized. In the simplest case they can be mapped by a *Bernoulli distribution* like Value (X) $= \{true, false\}$ with Value (X)$=\{v \in \Omega : f(v) = true\}$. The *sum rule* estimates the probability of the events α and β: $\mathcal{P}(\alpha \cup \beta) = \mathcal{P}(\alpha) + \mathcal{P}(\beta) - \mathcal{P}(\alpha \cap \beta)$. The distribution over one random variable like $prob(\alpha)$ is a *marginal* one and over several variables a *joint distribution*. A joint distribution is defined over a set of $\mathcal{X}=\{\mathrm{X}_1, ..., \mathrm{X}_n\}$. This holds for the discrete space. If we use the continues one, we need a non-negative, integrable density function like

$$\int_{Value(\mathrm{X})} prob(x)\, dx = 1 \text{ with } prob : \mathbb{R} \rightarrow \mathbb{R}, \tag{2.14}$$

which estimates the probability of a set of values, the *Probability Density Function* (PDF) . The according *Probability Mass Function* (PMF) is then

$$\sum_{x \in \mathcal{S}} f_{\mathrm{X}}(x) = 1. \tag{2.15}$$

Written in another way we get

$$\mathcal{P}(a \leq \mathrm{X} \leq b) = \int_a^b prob(x)\, dx \tag{2.16}$$

and the character of a *Cumulative Distribution Function* (CDF) becomes clear. A joint density function is then defined as the probability of any joint event over the variable of interest

$$\mathcal{P}\left(a_1 \leq \mathrm{X}_1 \leq b_1, ..., a_n \leq \mathrm{X}_n \leq b_n\right) = \int_{a_1}^{b_1} ... \int_{a_n}^{b_n} prob\left(x_1, ..., x_n\right) dx_1 ... dx_n \quad (2.17)$$

The *expectation* of an outcome, the mean value of X, can be estimated by $\mathbb{E}_{\mathcal{P}}\left(\mathrm{X}\right) = \int x * prob\left(x\right) dx$. The deviation of the values to the expectation is described by the *variance* $\mathbb{V}_{\mathcal{P}}\left(\mathrm{X}\right) = \mathbb{E}_{\mathcal{P}}\left(\left(\mathrm{X} - \mathbb{E}_{\mathcal{P}}\left(\mathrm{X}\right)\right)^2\right)$[110]. Further important parameters this work will refer to, like *covariance*, *correlation coefficient*, *standard deviation*, *entropy*, *mutual information* and *moments* of random variables can be found in the appendix A.2. The mostly used PDFs are the *uniform* and the *Gaussian distribution*. The *Central Limit Theorem* (CLT) states, that the PDF of a sum of random variables, seen under general conditions and with finite variance, converges to a normal distribution. This holds for many distributions in nature and also for many applications in the computer vision. For these approaches a combination with non-linear functions like the *Chi-, Rayleigh-* or *Maxwell distribution* is often used. In general, the PDF is unknown and must be guessed. Often used are fitting approaches with template PDFs to the data, where the problem of over-fitting exists. The most important characteristic of variables or events is the independence. An event α is *independent* of an event β, if $\mathcal{P}\left(\alpha \cap \beta\right) = \mathcal{P}\left(\alpha\right)\mathcal{P}\left(\beta\right)$. Probabilities can be interpreted as frequencies of events (*frequentist*) or as statement about the degree of belief (*subjective*). In this context I prefer the subjective interpretation, as I want to predict future events and to benefit from the connection between evidence and theoretical concepts. The probability theory is funded on set-theoretic basics and realized by general axioms. An interesting combination is given with the field of decision theory, which allows to make optimal decisions in face of uncertainty [23] and to predict the outcome of a new, unobserved event. To do this, a classifier is necessary, see sections 2.2.4 and 4.4.1.2.2 for more details. The probability of classifying to a wrong class, which can occur, if the range of values of the PDFs for more than one class overlap, is called *misclassification rate* while the opposing probability of being correct is given by

$$prob\left(correct\right) = \sum_{n=1}^{n} \int_{R_n} prob\left(x, C_n\right) dx, \quad (2.18)$$

with $\mathcal{R}_n$ as *decision boundaries* and $\mathcal{C}_n$ the possible classes. In the resulting approach a maximization of this equation should be used. If decisions can be weighted in their consequence of being chosen, independently of the fact if they were chosen correctly or not, the minimizing of the loss/cost function can help to improve the results to be more optimal. The *loss/cost function* is defined by

$$\mathbb{E}\left[L\right] = \sum_{n} \sum_{j} \int_{R_j} L_{nj} prob\left(x, C_n\right) dx, \quad (2.19)$$

with $\mathbb{E}$ as expectation, R as decision region, L the loss and C the given classes. A reformulation by the product rule gives a shorter equation $\sum_n L_{nj} prob\left(C_n \mid x\right)$,

that has to be minimized. The *reject option* can be used in cases where a wrong decision is worse than having no decision. This option is simply a threshold, which helps to decide, if the probability of the outcome is certain enough to be a base for a decision. With the help of these estimations the classification can be more robust and for the case of learning approaches able to be evaluated. Note, that the explanation here firstly describes the inference making by analyzing the training data and secondly assigning them to classes by the probability of the outcomes. To fuse these two stages, building a *discriminant function* like a *regression function* is possible [23]. Other mathematical frameworks of dealing with uncertainty are numerical approaches, *fuzzy logic* (vagueness), and *Dempster Shafer Theory* (evaluations of the degree of knowledge). Not only the uncertainty measurements or assumptions are modeled, also the different types of errors. Systematical variations are stable over the system and can be handled as offset. Optimally, they can be estimated by a calibration of devices, normally they can only be minimized. In contrast to them, the statistical error can not be estimated as it has a random occurrence. The repeatability can be massively degraded. Possible causes can be noise, too sparse experiments or random variabilities.

2.2.2.1 Conditional Probabilities and Bayes' Theorem

Using the constraint of considering only those instances of an event α for which the event β is given, leads to the definition of *conditional* probabilities, denoted as $prob\,(\alpha \mid \beta)$. The relationship between these events and their random variables X and Y are given by the *product rule* $prob\,(\mathrm{X}, \mathrm{Y}) = prob\,(\mathrm{Y} \mid \mathrm{X})\, prob\,(\mathrm{X})$. Note that $prob\,(\mathrm{X}, \mathrm{Y})$ is a joint probability, described by conditional dependencies. The next relationship can be formed by using the *symmetry property* and is usually used if uncertainty exists about the measurement parameters. This is the *Bayes' theorem* or *Bayes' rule*:

$$prob\,(\mathrm{X} \mid \mathrm{Y}) = \frac{prob\,(\mathrm{Y} \mid \mathrm{X})\, prob\,(\mathrm{X})}{prob\,(\mathrm{Y})} \text{ or } prob\,(\mathrm{X}) = \sum_{\mathrm{Y}} prob\,(\mathrm{X} \mid \mathrm{Y})\, prob\,(\mathrm{Y}.) \tag{2.20}$$

The general form is given by

$$prob\,(\mathrm{X} \mid \mathrm{Y}) = \frac{prob\,(\mathrm{Y} \mid \mathrm{X})\, prob\,(\mathrm{X})}{\sum_i prob\,(\mathrm{Y}, \mathrm{X}_i)\, prob\,(\mathrm{X}_i)}. \tag{2.21}$$

The Bayes' theorem is the basic tool for making inference in probabilistic expert systems [42]. Looking at the equation defines the *a priori information* ($prob\,(\mathrm{X})$, $prob\,(\mathrm{Y})$), the *a posteriori probability* ($prob\,(\mathrm{X} \mid \mathrm{Y})$) and the *likelihood* ($prob\,(\mathrm{Y} \mid \mathrm{X})$). The likelihood describes the uncertainty of data with a given parameter. By using the equation as query for $Y = y$, where y is a real value, Y becomes an evidence. By extending the product rule for the usage with three variables

$$prob\,(\mathrm{X}, \mathrm{Y}, \mathrm{Z}) = prob\,(\mathrm{X} \mid \mathrm{Y}, \mathrm{Z})\, prob\,(\mathrm{Y}, \mathrm{Z}) = prob\,(\mathrm{X} \mid \mathrm{Y}, \mathrm{Z})\, prob\,(\mathrm{Y} \mid \mathrm{Z})\, prob\,(\mathrm{Z}) \tag{2.22}$$

which can be generalized to

$$prob\left(\mathrm{X}_1, \mathrm{X}_2, \ldots, \mathrm{X}_n\right) = prob\left(\mathrm{X}_1 \mid \mathrm{X}_2, \ldots, \mathrm{X}_n\right) \ldots prob\left(\mathrm{X}_i \mid \mathrm{X}_{i+1}, \ldots, \mathrm{X}_n\right) \ldots prob\left(\mathrm{X}_n\right). \quad (2.23)$$

To find the best decision the search for the *Most Probable Explanation* (MPE), also called *Max A Posteriori* estimation (MAP) is defined by

$$MAP\left(\mathrm{X} \mid \mathrm{Y}\right) = argmax_{\mathrm{X}} prob\left(\mathrm{X} \mid \mathrm{Y}\right). \quad (2.24)$$

An alternative are risk functions where the definitions are dependent on the chosen distance metrics. The Bayes statistic has therefore some differences to the classical statistics [111]. The first enhancement is the quantification of the plausibility of parameters, then the modeling of unknown parameters as random variables and finally the ability to build hypothesis and predictions of parameters. As an enhancement, the usage of time steps and spatial dependencies leads to stochastic processes.

2.2.2.2 Stochastic Processes w.r.t. Semantical Networks

The time-dependent ordering of randomized processes forms a *stochastic process*. If we assume that Ω is given as vector space with $w \in \Omega$, then the definition of a stochastic process P follows by

$$P = \mathrm{X}\left(w, t\right) \text{ with state space } V = \left(\Omega \mathrm{x} T\right), V \rightarrow \mathbb{R}^n \text{ and } T \in I \subset [0, \infty[\,. \quad (2.25)$$

Due to their variability and multitude of restrictions there exists a wide range of types of processes [133, 216]. The classification is normally done by the differentiation of discrete or continuous models for the spaces of time or values. Beside *Lévy-* (like *Wiener-* or *Poisson-*) processes and *Gaussian*-processes, the *Markov* processes have gained a wide field of applications in the pattern recognition field. Markov processes define probabilities for the state transitions in a stochastic process. State transitions are given by

$$prob_{ij}\left(t, t+1\right) = P\left(\mathrm{X}_{t+1} \mid \mathrm{X}_t\right) \text{ with } i, j \in V. \quad (2.26)$$

In general, $\sum_{j=1}^{n} prob_{ij} = 1$ holds. The state transitions can be written as a quadratic *transition matrix*

$$prob_{ij}\left(t\right) = P = \begin{pmatrix} prob_{11} & prob_{12} & \cdots & prob_{1N} \\ prob_{21} & prob_{22} & \cdots & prob_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ prob_{N1} & prob_{N2} & \cdots & prob_{NN} \end{pmatrix} \quad (2.27)$$

with $prob_{ij} \in [0, 1]$ for $i, j = 1, ..., N$ and still $\sum_{j=1}^{n} prob_{ij} = 1$, to form a stochastic matrix. With this matrices per each state, the overall probability can be written as

$$P\left(\mathrm{x}_1\right) * P\left(\mathrm{x}_2 \mid \mathrm{x}_1\right) * P\left(\mathrm{x}_3 \mid \mathrm{x}_2\right) * \cdots * P\left(\mathrm{x}_n \mid \mathrm{x}_{n-1}\right) = P\left(\mathrm{x}_1\right) \prod_{i=1,j=2}^{n-1,n} prob_{ij}. \quad (2.28)$$

Markov chains are a specific case were the spaces Ω and I are discrete [90]. These

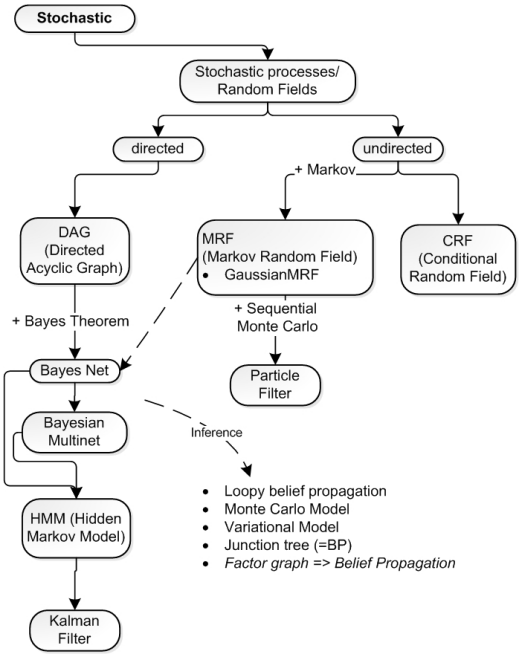

Figure 2.8: Thematic overview of the interesting relationships in the field of stochastic processes.

chains have an order, which describes the number of states in the past, which will influence the estimation. The most interesting approach to this work is the usage of graphical, stochastic, Markov models like *Petri Nets* or *Markov Fields* (MF). While Petri Nets grow very fast and result in very complex, detailed models, the Markov nets provide the integration of a probability framework and break down the inference strategy to local estimations. See figure 2.8 for a brief overview of this topic of the stochastic view on Markov Fields. The field can be divided in *undirected* and *directed* graphical models. Beginning with the fully undirected case, the most general networks are given by the *Markov Random Fields* (MRF). The definition is as follows: a graph $\mathcal{G}$ over a set of variables $\mathcal{V}$ and a set of edges $\mathcal{E}$ with $\mathcal{E} \subset \mathcal{V}x\mathcal{V}$ is an undirected one, if $\mathcal{E}$ only contains undirected edges $(i,j) \in \mathcal{E}$. The neighborhood set of i is given by $\delta_i \equiv \{j : j \sim i\}$. MRFs are then graphs with a joint distribution $prob\,(x) \equiv prob\,(x_1, \ldots, x_n)$ which can be defined locally by conditional distributions [89]

$$prob\,(x_i \mid x_{i-1}) = prob\,(x_i \mid x_{\delta_i})\,, i = 1, \ldots, n. \tag{2.29}$$

If $prob\,(x) > 0$ the equation becomes

$$prob\,(x) = prob\,(0) \prod_{i=1}^{n} \frac{prob\,(x_i \mid 0, \ldots, 0, x_{i+1}, \ldots, x_n)}{prob\,(0 \mid 0, \ldots, 0, x_{i+1}, \ldots, x_n)}, \tag{2.30}$$

as it is determined by its local conditional distributions. This equation satisfies the following conditional independence properties

- *pairwise Markov property*: an edge between two nodes is absent in the graph if X_i, X_j are conditionally independent

$$\mathrm{X}_i \perp \mathrm{X}_j \mid \mathrm{X}_{\setminus(i,j)} \tag{2.31}$$

- *global Markov Property*:

$$\mathrm{X} \perp \mathrm{Y} \mid \mathrm{Z} \tag{2.32}$$

  for $\mathrm{X}, \mathrm{Y}, \mathrm{Z} \in \mathcal{V}$ so that Z separates X from Y

- *local Markov Property*:

$$\mathrm{X}_i \perp \mathrm{X}_j \mid \delta_i \tag{2.33}$$

  for all $j \in \mathcal{V} \setminus (\delta_i \cup \{i\})$

- global Markov Property⇒ local Markov Property
  ⇒ pairwise Markov property[195].

Partially directed graphs lead to *Conditional Random Fields* (CRF), which equally to MRFs have no causal interpretation. A directed graph without cycles is called an DAG, as mentioned in section 2.1.2. DAGs can be enhanced to be a causal network with the usage of probabilities (belief network), for example a *Bayesian network* (BN) or a *Hidden Markov Model* (HMM)-Field. The main question is, which data should be modeled. Bayesian networks are directed like a DAG and have resulting restrictions, like no cyclic dependencies. Nodes represent the random variables and the edges are the conditional dependencies. See Figure 2.9 for an introduction of the basic structure. More details on that topic follow in section 2.2.3.
A generalization of the BN are *Influence Diagrams* (ID), which can combine several networks, mixed BNs and flexible models. HMMs are the simplest form of a dynamic BN. HMMs extend the Markov model to model unseen (non-physical) variables as hidden variables. The hidden variables can thereby be estimated by the probabilistic function of the state, which can be embedded in a stochastic process like the Markov model [55]. An access to this field gives [162]. Stochastic process methods can also

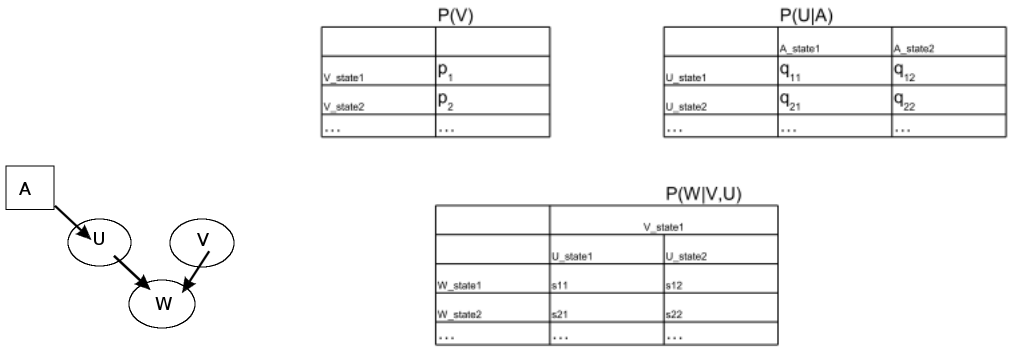

Figure 2.9: Simple example of a Bayesian network structure. The nodes represent conditional probability tables.

be used for finding parameters towards regions where we expect the solution. *Boltzmann learning* (with strategies of *Simulated Annealing*) and *Evolutionary methods* like *Genetic Algorithms* have to be mentioned. For BN and MF sampling methods can help to simulate the measurements on guessed distributions, e.g. *Monte Carlo methods* (random sampling) like *Markov Chain Monte Carlo* (MCMC) sampling, *Particle filter* or *Importance sampling*. Particle filters gained to be a very good alternative to *Kalman Filters* (KF) or *Uncented Kalman Filter* (UKF) [228], as they reach the Bayesian optimal estimate.

2.2.3 Bayesian Networks

Where ML estimations fail, e.g. in small data sets or missing data conditions, the Bayesian statistic can deal with initial guesses called a priori information. The basic functionality of the Bayesian statistics is the reasoning about a random variable with implied distribution if a precedent random variable is given.
BNs are directed graphs which can model a knowledge domain completely under uncertainty. The explicit estimation of the joint probability of such a complex knowledge base that is needed in this context is too expensive to compute. Even if we regard a *Bernoulli* distribution over n variables the effort would be exponential by 2^{n-1} values. Similar to the real world, the relations in the network model causal dependencies. By using the Bayes rules, the network can model high-dimensional probability distributions by local dependencies, which leads to a linearized effort of $2n + 1$.
A BN $\mathcal{G}$ is a DAG with nodes that represent random variables with conditional dependency assumptions. Given pa_{x_i}, each node x_i is independent to all other nodes in $\mathcal{G}$, except its parents. Not all distributions can be modeled as BN. If $\mathcal{I}(\mathcal{G})$ is the set of all conditional independences of $\mathcal{G}$, $\mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(\mathcal{P})$ must hold. That means, $\mathcal{G}$ is an *i-map* for $\mathcal{P}$ and can be factorized by using the chain rule to satisfy 2.23. If $\mathcal{I}(\mathcal{G}) = \mathcal{I}(\mathcal{P})$ the map is perfect and the independences can be directly transformed (*p-map*). Nodes in a BN represent the domain variables with a set of mutually exclusive states. The causal links between them can demonstrate serial, divergent or convergent connections. *Serial* relations model dependencies between several simply-connected nodes. If three nodes are connected in a chain, the reasoning will not be possible, if the state of the embedded node is not known, which refers to *d-separated* nodes. If the state is known, the inference can be passed to the last node in the chain. *Diverging* connections refer to a generalization node which d-separates it's children in the instantiated case and *converging* connections use the principle of *explaining away*, which means, that parents become dependent if their shared child gets instantiated. The instantiation can be done by exact knowledge, called *hard* evidence or otherwise by *soft* evidence. If two nodes are independent although they are dependent by the same parent node, they are defined as *d-separated*. Distributions exist which can have influences on both of the nodes, even if they are independent. To ensure the correctness of the d-separation, three properties must be additionally fulfilled. The *soundness* states, that the i-map must hold if the observed independence assumption is supported by the underlying distribution. The second property is *completeness* which calls for the condition, that all possible independences have

to be mapped. *Faithful* are distributions, if whenever $(x_i \perp x_j \mid x_k) \in \mathcal{I}(P)$, then $d - sep_{\mathcal{G}}(x_i; x_j \mid x_k)$ holds [207, 222, 99, 110]. By definition, nodes without parents need to have an a priori value $P(x_1 = i)\, \forall i$, whereas nodes with parents have to have specified conditional probabilities $P(x_1 = i \mid x_2 = j, x_3 = k)$, $\forall i, j, k$, which are representable in conditional probability tables $P(x_1|x_2, x_3)$. A special node type to handle a big number of parents for a node is *Noisy-Or*, which's state becomes more probable the more parents have satisfying instantiations. The neighborhood of a node is called *Markov blanket* in BNs. All other basic definitions for graphs (see 2.1.2) hold for BNs. The advantage of a BN is the definition of the joint probability by the repeated application of the product rule (see 2.23). By doing so, it is possible to reduce the high-dimensional joint distribution of the whole network to a product of local conditional distributions (used chain rule)

$$prob(x_1, \ldots, x_K) = prob(x_K \mid x_1, \ldots, x_{K-1}) \ldots prob(x_1 \mid x_2)\, prob(x_1) \tag{2.34}$$

with K nodes. This recursive factorization holds in complete DAGs, more generally written as

$$prob(x) = \prod_{k=1}^{K} prob(x_k \mid pa_k)\, [23]. \tag{2.35}$$

BNs should be used to increase the performance of complex conditional dependencies. BNs can be used by assigning four different query tasks. Given the model of the observed domain, a prediction can state what might happen. If evidence values are given, the estimation of the a posteriori probability gives knowledge about the dependent events or about the scheme of events. If a posteriori information is given, the question of the most likely source of the event can be answered and the domain model can analyze decision making procedures. The most relevant query is the one if some evidence can be provided. In this case, two topics deal with the usage of BNs: the probability updating (propagation) and the inference schemes to estimate the distribution of the unknown variables. Both will be handled in the following sections.

2.2.3.1 Knowledge Representation

Knowledge based systems differentiate between the knowledge base and the inference strategy. If the knowledge base is provided by a human expert of the topic, these systems are called expert systems. Expert systems have been developed since the end of the 1960's. An expert has some characteristics which make the applications very successful:

- An expert uses heuristic knowledge, own experience and general knowledge.
- Experts have superior knowledge in at least one topic.
- Experts can solve problems in face of imperfect and uncertain knowledge.
- Experts are expensive and rare, but a single expert is often not sufficient enough.

The systems have a modular layout, the knowledge is given declaratively by the representation of an expert. For specific applications a temporary case-sensitive instance of the base can be built, which is done by a knowledge processing unit. There

the problem solving strategy is chosen and used for the deduction. Furthermore the systems have input/output procedures.
Knowledge can be conveyed by logic- or rule-based information. Decision trees are an example of sequences of questions which do not separate knowledge base and inference engine. Logic-based approaches define propositional and predicate logics, formulas and relations. Rule-based systems use conditional clauses to describe "if condition then action" conditions. They are often represented by graphical models like semantical or neural networks, but there also exist hybrid approaches. Intelligent behavior is the result of being able to conclude from a knowledge base. Thus, the inference strategy is the main aspect which decides of the success on a system. Peirce [20] differentiates the types of strategies into deduction (from general to specific cases), induction (from specific to general cases) and abduction (new knowledge by observation). In combination with them, one can use the reasoning with uncertain data. Due to the problem in deduction methods, that the number of rules can only increase (non-monotonic reasoning), the usage of probabilities can weight and mask rules. Rule-based systems were the earliest approaches to represent expert systems. As first of all their limitations when dealing with uncertainty were shown. They can grow enormously and have to deal with redundancy and consistency, resulting in reasonings which can be too complex [42]. The probability theory is considered in section 2.2.2 in more detail. The frequentist interpretation [217, 182] leads to classical statistical inference by Newman and Pearson [140] and for the subjective interpretation one the most known occurrence is the Bayesian interpretation [42].
When uncertainty is represented by a probabilistic framework and gives the possibility for subjective interpretations, the Bayes theorem leads to a powerful belief propagation. One limitation is the disability to model new cases. Therefore normative rule-based systems [85] are more general in the problem definition and do not model the knowledge base of an expert anymore but model the domain knowledge. Normative reasoning under uncertainty uses the classical probability theory as mathematical framework [99], [98]. To recieve the model of a whole domain, an often used approach is *case-based reasoning* (CBR) which uses a data base of cases. This approach is able to grow in the domain knowledge whereas the rule-based systems, or more precisely, the rule-based networks can practically not model the full knowledge base of a domain. A BN is a declarative representation, because it separates the models of knowledge and reasoning.

2.2.3.2 Evidence Propagation and Exact Inference

Inference in BNs is the estimation of a distribution for an unknown variable. While update mechanism can be done with low effort, the inference in BNs is proven to be *NP-hard* [45], as for each node the a posteriori probability via Bayes rule has to be estimated. To be able to make efficient inference on graphs, many graphical operations are available. Efficient approaches to solve the problems are the *Polytree Algorithm*, *Clustering methods*, *Conditioning*, *Arc Reversal*, *Symbolic methods* and *Differential methods* [71]. They will be introduced in the following.

2.2.3.2.1 Belief propagation Only one subset of BNs, namely polytrees, can be solved in linear time. Very early, Pearl [226] applied the variational method belief propagation as exact inference algorithm. The procedure, which is a special case of the general sum-product algorithm, works on different graphical models like MNs or factor graphs, but for exact inference a loop-free structure is needed. *Pearl's Belief propagation* is a message-passing system which can push the data through both: the a priori and the a posteriori probabilities. The node N which shall be estimated, splits the network into two parts, in which the passed information is handled as messages. The upper part will be updated by *forward passing* with π-messages to the parents $pa_1, \ldots, pa_m$ and the lower part will get new information by *backward passing* with λ-messages to the children $ch_1, \ldots, ch_n$. Evidence is e with the evidence at ancestors e^+ and the evidence at decendants e^-. For a Node X holds:

$$\mathcal{P}(X \mid e) = \alpha \lambda(X) \pi(X) \tag{2.36}$$

with α as normalizing factor and

$$\lambda(X) = \mathcal{P}\left(e^- \mid x\right) = \prod_{i=1}^{n} \lambda_{ch_i} \tag{2.37}$$

and

$$\pi(X) = \mathcal{P}\left(x \mid e^+\right) = \sum_{i=1}^{m} \mathcal{P}(x \mid pa_i) \pi(pa_i). \tag{2.38}$$

The following λ- and π-messages to update the children and parents of X are:

$$\pi(ch_i) = \frac{\mathcal{P}(X \mid e)}{\lambda_{ch_i}} \tag{2.39}$$

and

$$\lambda(pa_i) = \beta \sum_X \lambda(X) \sum_{k \neq i} \mathcal{P}(X \mid pa_k) \prod_{k \neq i} \pi(pa_k) \tag{2.40}$$

with β as normalizing factor [26].
Many optimizations exist which simplify the underlying graph (see next paragraph) or extensions of the procedure to update latent variables (*max-sum* algorithm) or loops (*loopy belief propagation*).

2.2.3.2.2 Clustering Algorithms If no polytree is given, procedures to transform the graph into common structures have been developed. A similar message passing procedure is then applied on the clusters. The message passing strategy can be reformulated as junction tree algorithm. If the graph is a singly-connected one (a tree), the junction tree can be easily given by forming clusters like $pa(N) \cup \{N\}$ for a node N and if $pa(N) \neq \emptyset$. All other directed graphs firstly have to be transformed into an undirected one by a moralization procedure to a moral graph. Doing this, all edges are handled as undirected and additional undirected edges are set between nodes with a common child. The moral graph is the basic graph structure for junction tree approaches. It follows a triangulation step to find expanded sets

of variables with not more than three variables. On the triangulated (or chordal graph) optionally a variable reduction can be performed (bucket elimination) to avoid redundancy within the estimations. The rearrangement of choosing the nodes to be the maximal cliques and the links to be a connection of cliques with common variables leads to the join tree (or cluster tree). The junction tree (or clique tree) is then given by a condensed join tree, so that no subsets exist anymore. After these graphical restructuring methods, a sum-product like message passing is done [42, 23].

2.2.3.2.3 Conditioning For loopy structures and for applications with low computational space requirements, an alternative to clustering algorithms has been provided. The idea of conditioning is the reduction of the mass of cliques which have to be handled in clustering approaches. The definition of subsets like $C \subset X \backslash E$ allows to break down the net into smaller, independent pieces. These subnetworks are called *cutsets* and for each node $X_i \in X \backslash (C, E)$ the a posteriori is estimated by:

$$prob\,(x_i \mid e) = \sum_{c \in \mathcal{P}(C)} prob\,(x_i \mid c, e)\, prob\,(c \mid e)\,. \tag{2.41}$$

While Pearl's algorithm can be viewed as a global conditioning approach, improvements have been introduced by Peot, Schachter and Diez (local conditioning) [23].

2.2.3.2.4 Arc Reversal While variable elimination multiplies all distributions referring to a variable and sums useless variables out of the associated product, the arc reversal approach reduces the net by iteratively applying the Bayes rule. The arcs v_i, v_j are reversed for each child v_j of v_i,by

$$prob\,(v_i, v_j \mid A_i) = prob\,(v_i \mid A_{j-1}) * prob\,(v_j \mid P_j)\,, \tag{2.42}$$

$$prob\,(v_j \mid B_j) = \sum_{v_i} prob\,(v_i, v_j \mid A_j) \tag{2.43}$$

and

$$prob\,(v_i \mid A_j) = \frac{prob\,(v_i, v_j \mid A_j)}{prob\,(v_j \mid B_j)} \text{[33]}. \tag{2.44}$$

2.2.3.2.5 Symbolic and Differential Methods Symbolic probabilistic inference approaches (SPI) allow the replacement of numeric values by functions of the parameters. Analysis on the parameter space can be done without expensive recalculations by just replacing the symbol with relevant data. Both kinds of data, numeric and symbolic, can be mixed in the same underlying BN to provide the best benefit. Differential methods calculate the polynomials in the network and try to do the estimation by partial derivatives referring to the chosen variables [46].

2.2.3.3 Evidence Propagation and Approximative Inference

A classification of this topic is normally done with four categories: *Stochastic sampling*, *Model Simplification*, *Search-based* and *Loopy propagation* [71]. Approximative approaches are needed, if the models have grown too large and exact inference is

not manageable anymore. As a preprocessing, relevance-based algorithms ([54, 119] help to define the graph variables that necessarily need to be updated. Only the first category of stochastic sampling procedures is discussed in more detail, as their procedures will be used to infer the BNs used in this work. For the building of cliques in the propagation process, the conditional probability tables are necessary. If the cliques are very large, a simulation of the tables can help to approximate the values. The sampling procedures can be divided into importance sampling methods, which handle the variables independently and Markov Chain Monte Carlo methods. The idea of the sampling procedures is to estimate the expectation of a function correspondent to a given probability function to be able to predict the a posteriori probability for some variables. The expectation for the continuous space is given by

$$\mathbb{E}[f] = \int f(z)\, prob(z)\, dz. \tag{2.45}$$

After the sampling from a distribution $prob(z)$, the samples z can be used to approximate the expectation by

$$f^{'} = \frac{1}{L}\sum_{l=1}^{L} f(z_l). \tag{2.46}$$

The aim is the reduction of the variance

$$\mathbb{V}\left[f^{'}\right] = \frac{1}{L}\mathbb{E}\left[(f - \mathbb{E}[f])^2\right]. \tag{2.47}$$

Problems occur, if the samples are not independent or if the sample size grows too large.

2.2.4 Classification

A classification is attended to split the feature space into interesting classes. A feature space is defined over the estimated parameters of the training data set. Figure 2.10[4] shall introduce to that topic. The wide range of classification approaches deal with the estimation of class distribution to characterize the optimal boundaries. In classification, a *pattern vector* $\mathbf{x}$ is defined over a p-dimensional vector space Ω which can be spanned by p features. A *classifier* takes the input vector $\mathbf{x}$ and finds the mapping function to label the value for a discrete class $w_k \in C$ with $k = 1, ..., n$ the number of classes and has to be derived by a set of training data $\mathcal{X}$ of labeled classes [39]. The class data form regions in Ω, their bounding surfaces are called *decision boundaries*. The evaluation of the data is normally done by their distributions in Ω. Often used characteristics are variances, the distances of their means and their covariances. To estimate the optimal distribution of the data several mathematical approaches can be used: statistical like probability distributions (parametric / non-parametric) or regression, discriminant functions and graphical models for exact boundaries, mainly depending on the data being labeled or not, what refers to the supervised and unsupervised dataset. This section will neglect

[4] illustrations with copyright by Christopher M. Bishop

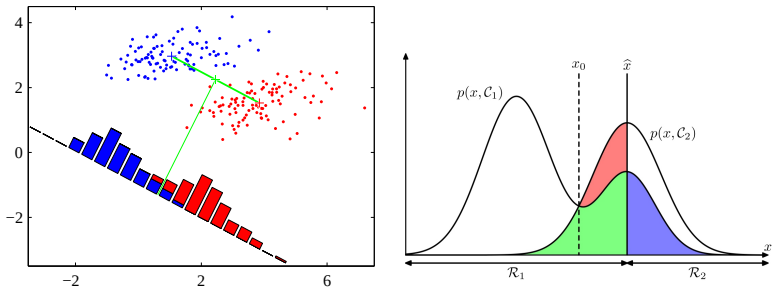

Figure 2.10: Left: A feature space with plotted parameter estimations is shown with blue and red dots. As illustration, the according distribution of the classes is given as histogram. Right: Distributions for similar histograms are shown with overlapping regions [23] which symbolize the problematic regions and therefore error sources as introduced in section 2.2.2.

structural approaches and NN, because they build an unique topic which is not addressed by this work. The section gives an overview of classification approaches and the placement of Bayesian methods.

2.2.4.1 Supervised

Linear Discriminant Analysis (LDA) for multi-class problems can be composed in different ways. Either, the more simpler case of a two-class problem can be used to build an *ensemble classifier*. Each classifier discriminates each value between two classes and a voting scheme estimates the overall outcome. Or each class will be trained for one classifier. With a set of training data $x_1, ..., x_n$ with given classes $w_1, ..., w_n$ a discriminant function g_k can be defined, using the weight vector w^k and a threshold w_0^k for the k^{th} classifier in that way

$$g_k(\dot{x}) = \left(w_0^k\right) + \sum_{n=1}^{n} (w_k)^T x_n \begin{cases} > 0 \\ < 0 \end{cases} \quad \text{so that } x \in \begin{cases} w^k \\ w_1, ..., w_{k-1}, w_{k+1}, ..., w_n \end{cases} \tag{2.48}$$

If $\dot{x} = \{1, x_1, ..., x_n\}^T$ and $\dot{w} = \{w_0, ..., w_n\}^T$ than we can shorten to

$$g_k(\dot{x}) = \dot{w}^T \dot{x}. \tag{2.49}$$

The weight vector refers to the slope of the boundary, whereas w_0^k is the distance to the origin. If no clear correspondence can be set, there is no single class with a positive value for g_k^x, and a voting scheme has to be found. The third solution is to define n discriminate functions and the mapping of x to a class w_i if $g_i(x) = max_j g_j(x)$ [224]. As we want a weight vector which assigns as many values as possible to the correct class, a minimization of the misclassification error is essential. Such a criterion can be the *Perceptron* criteria,

$$J_P\{\dot{w}\} = \sum_{\dot{x} \in X(\dot{w})} -\dot{w}^T \dot{x} \text{ with } X(\dot{w}) \text{ as misclassified data,} \tag{2.50}$$

with a gradient-based estimation or the *Fisher's criterion*, which weights the misclassified examples due to their distances to the decision boundaries

$$J_F = \frac{\left|\dot{w}^T (m_1 - m_2)^2\right|}{\dot{w}^T \; S_W \dot{w}} \tag{2.51}$$

with m_1 and m_2 as group means and S_W as within-class covariance matrix given as

$$\frac{1}{n-2} (n_1 \Sigma_1 + n_2 \Sigma_2) \tag{2.52}$$

with Σ_1, Σ_2 as likelihoods of the covariance matrices and n_i samples. Euclidean metrics like *least mean squared error* or *Mahalanobis distance* can be equally used as a cost function. If we assume Σ_1, Σ_2 not to be equal, this leads to *Quadratic Discriminant Analysis* (QDA). The discriminant analysis can not deal with overlapping decision boundaries and needs normal distributed values. But it can not only separate, it can also characterize the training sets and optimize the feature space. Approaches to do this are the LDA itself furthermore the *Principal Component Analysis* (PCA), the *Independent Component Analysis* (ICA) and *factor analysis* which are explained in the unsupervised section 2.2.4.2. The LDA uses as prior information the densities of the data. This can be parametric or non-parametric suggestions or estimates. If we generalize the LDA to piece-wise linear discriminate functions, the nearest vector in the space Ω can be chosen by the nearest-neighbor approach which tessellates the feature space. LDA can be stated as a special case of a regression method, as binary LDA is equivalent to linear regression [230].
Regression does not map an input directly to an output value, but assumes conditional dependencies between input and output. Regression is more flexible than LDA and does not restrict the data to normal distributions. In contrast to the LDA approaches, regression can not handle categorical data well but serves as an estimate of conditional expectation. The model of the regression has the given form with available generalizations to non-linear and logistic regression [77]:

$$f(\mathrm{X}) = \beta_\mathrm{o} + \sum_j \mathrm{X}_j \beta_j \tag{2.53}$$

with β_j as coefficients learned from the trainings data, X_j as input variables and the assumption that the regression function $E = (\mathrm{Y} \mid \mathrm{X})$ is linear. A famous approach is the estimation of β_j by the *least squares approach*, minimizing the residual sum of squares

$$RSS(\beta) = \sum_i (y_i - f(x_i))^2 = \sum i \left(y_i - \beta_0 - \sum_j x_{ij} \beta_j \right)^2 . \tag{2.54}$$

With many classes and few dimensions, the linear regression model often is much worser than the LDA (suppression of classes). From the statistical viewpoint, a classifier assigns the vector x to the most likely class, which refers to the maximum posterior probability. An optimal classifier which will get the minimum error is the

Bayes optimal classifier. If we have for two classes the known values for $prob\,(w_1 \mid x)$ and $prob\,(w_2 \mid x)$ and the misclassifying factor is for both the same, we get

$$argmax_{v_k} \sum_{h_i \in H} \mathcal{P}\,(v_k \mid h_i)\,\mathcal{P}\,(h_i \mid D) \tag{2.55}$$

with v_k as possible classification, and H all possible hypotheses, which is an estimation over all hypotheses weighted by their posteriors. As this is very expensive to compute, an often used approximation by using the Bayes theorem (see section 2.2.2.1), is the *Naive Bayes classifier* which reduces the estimation to a calculation of the densities per variable a_i (conditional independence)

$$argmax_{v_k} \mathcal{P}\,(v_k) \prod_i \mathcal{P}\,(a_i \mid v_k)\,. \tag{2.56}$$

This classifier is cheap and often used for high dimensional spaces. Finding $prob\,(x \mid w)$ is the problem of fitting a density on a data sample. We need the probability that a value is assigned to a special class, what means, that each class has to be described by a PDF. Using a Gaussian density $\mathcal{N}$ gives

$$prob\,(x \mid w) = \mathcal{N}\,(x; \mu; \Sigma) = \frac{1}{(2\pi)^{\frac{p}{2}}\,|\Sigma|^{\frac{1}{2}}} exp\left(-\frac{1}{2}\,(x-\mu)^T\,\Sigma^{-1}\,(x-\mu)\right) \tag{2.57}$$

with μ as mean of class w and Σ as covariance and $|\Sigma|$ as determinant. Because the values μ and Σ can be estimated by e.g. maximum likelihood the classifier is named *plug-in Bayes classifier.* Many densities can be used in this context (see the appendix A.2.4 for a listing), even non-parametric like the *Parzen* density. Using a window function, this classifier is also called *Parzen Window.*

2.2.4.2 Unsupervised

Trees can represent linear and non-linear solutions, see the sections 2.1.2 and 4.4.3 for the generalization and continuation into the topic of semantical nets like Bayesian networks.
Where the supervised approaches try to examine the structure in the feature space, the unsupervised case tries to define describable structures. An unsupervised training data set has no labels and therefore no class considerations. The aim of the unsupervised approaches is to find clusters in the data which refer to classes. The first step is to find clusters and the second to describe them by a fitting density function. In literature, the unsupervised case is often only called *clustering.* See figure 2.11 for a brief summary of the hierarchical and partitioning approaches used in this topic. The approaches differ by their characterization of clusters (by their mean, their covariances). Beside the density estimation, the clustering of the data can be used to provide an optimized set of features (redundancy reduction, definition of dominant features). This can be done, also as improvement in the linear, supervised case, by the methods PCA, ICA and factor analysis which refer to the *statistical projection methods.* The projection maps the features into a lower dimensional space and increases the independence between the features. Because of the difficulty to

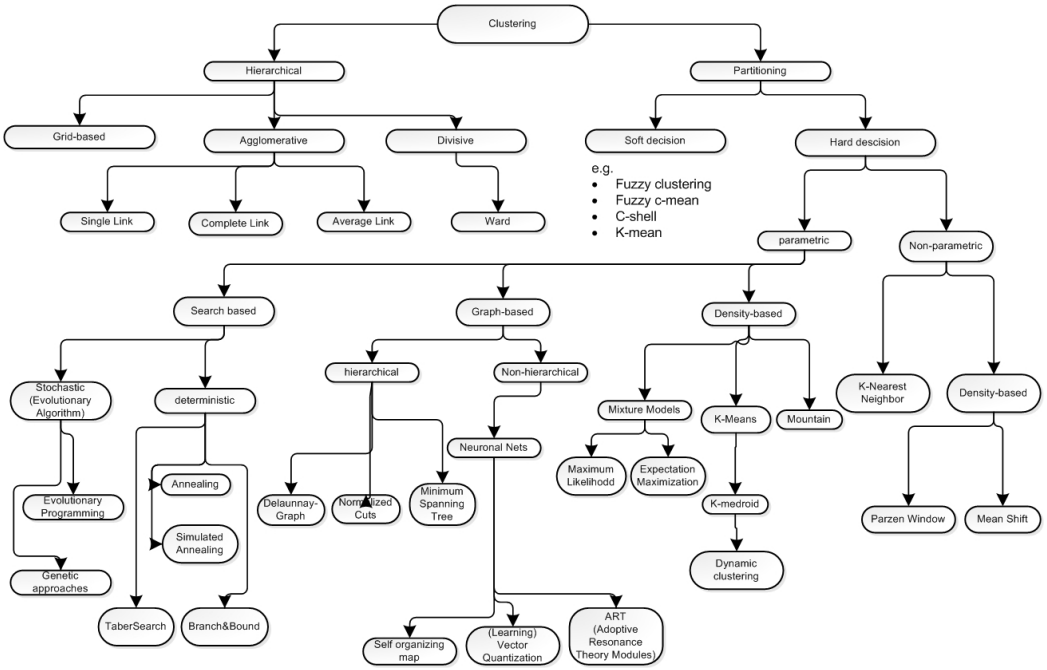

Figure 2.11: Overview of clustering algorithms.

define structures or to pretend the correct mathematical model which differs mainly on each training set, an often used approach is to combine different classifiers. Very well known are *bagging* and *boosting* approaches. The bagging collects different features into a group to optimally describe an object whereas the boosting approach trains the classifier incrementally on the data set [23][127].
An independent topic are Support Vector Machines (SVM), which are often used in these cases. SVMs map the pattern vector to a higher dimensional feature space where the discrimination can be done by hyperplanes. Hyperplanes are then defined by their distances to the coordinate origin and their positive normal vector. Points, defined on the separating hyperplanes, are the so-called *support vectors*. They became very popular because of their ability to optimize the sets of values for training and their usage of linear or non-linear kernels. The simplest SVM is a LDA model like eq. 2.48, these models can be used for linearly separable data sets. To maximize the margins between the hyperplanes, the general optimization strategy is done by *Lagrange multipliers* and with the help of the dual form of the Lagrangian. In the non-separable case, the models can be built more robust to outliers and with softer constraints for the margins with so-called *slack variables*. Special formulations lead to an independence of the estimation from the data set. The estimations only refer to the support vectors and their similarity to the new object and the error measurement. Due to this, this inner product $x^T x_i$ can be replaced by another similarity like a *Gaussian kernel function*, e.g.

$$K(x, x_i) = exp\left(-\frac{|x - x_i|^2}{\sigma^2}\right) \tag{2.58}$$

Many other kernel functions are possible. As a result of this section, discriminating functions can be chosen to be linear, quadratic, logistic, polynomial or represented by radial basis functions, multilayer perceptrons or non-parametric methods [224], [39].

Chapter 3

Augmented Reality

Contents

3.1 Terminology of Augmented Reality

As an introduction, this section explains the term *Augmented Reality* (AR) and shows the main applications in recent years. The following sections will illustrate the idea of an enhanced reality, the history of the usage and the actual research in this field. This work starts with a short explanation of the vision of the technology and it follows a clear definition of the terminology. Important applications of this topic, their drawbacks and open research points are shown.

3.1.1 Definition of Terms

A *Virtual Reality* (VR) can represent a synthetic or the real world, recreated in a pure simulated form. The usage and access to this data is restricted to a virtual environment. The world can be used to play, to learn and practice processes and to explore new knowledge. A VR is interactive, manipulative and in real-time. To achieve a high immersion (the most possible realistic impression) the input and output devices are especially developed for this context. Stereo output technologies are based on the human vision system and gloves with haptic sensors are used as input devices. In the beginning there was an open discussion about the definition of the concepts of Virtual and *Mixed Reality* (MR). The technology of AR was developed in the context of the development of *Head Mounted Displays* (HMD). The definitions of the term *Augmented Reality* were closely related to the HMDs and later to the display type for which first applications had been developed [13]. The first HMDs were developed by Ivan Sutherland (Massachusetts Institute of

Technology) and Raymond Goertz (Argonne National Laboratory) in [201][56] and in the eighties the first military applications had been used. In 1994, Milgram et al. explained the concept of AR in the context of a *Reality-Virtuality* (RV) continuum [137].

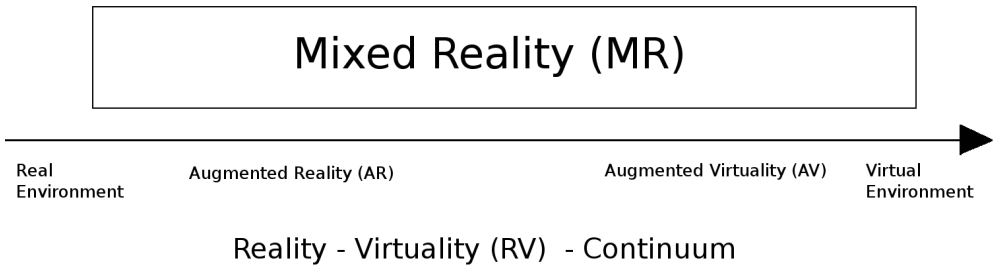

Figure 3.1: Reality-Virtuality Continuum on the base of Milgram [138].

The scheme describes different levels of realism. On the left side the real environment is situated and the more virtual data is added the more the term *Virtual Reality* (VR) is used. The parts between real and virtual environment should be called *Mixed Reality* (MR) or *Augmented Reality* (AR) in the use of today. In this way Milgram uses AR to enhance real scenes with computer graphics. The definition is not longer restricted to visual content [3] as haptic and auditory enhancements are allowed now. In 1997, Azuma [12] constraints the definition to the following three points: AR combines real and virtual, AR is interactive in real-time and AR data is registered in 3D. Azuma mentioned that AR would be an example of *Intelligence Amplification* (IA), being described as

> *"using the computer as a tool to make a task easier for a human to perform",[29]*

allowing a much more general definition. Today, AR is a term in broad usage, as more and more applications increase the range of possibilities. Every kind of data displayed into the real environment is an augmentation. This is not restricted to 3D registered data, as overlays are allowed as well. For this reason the field of AR is very large and a classification of technologies and applications is challenging. In [137] an extension of the *Taxonometry of MR* is introduced. Milgram defined more dimensions of immersion to classify MR displays and created the taxonometry with the help of the following terms: *Extent of World Knowledge* (EWK), *Extent of Presence Metaphor* (EPM) and the *Reproduction Fidelity* (RF). This kind of definition only refers to the display types used and therefore can only be regarded as a subtopic of AR. There are several formal definitions of the term AR [22], but as a conclusion, AR does not suppress the reality entirely, as VR does. It is the mixture of real and virtual data, we have to deal with.

3.1.2 History of AR Systems

The first ARS was developed by a cinematographer named Morton Heilig during the late fifties. His device is the *Sensorama*, which is known today as the first VR device. He used optical sensors, sound, vibration and smell for immersion [78]. The next step in the development was a HMD, tracked by mechanical and ultrasonic sensors [201], [221]. For the usage in real-time modus, the device could only display objects as wireframe. In the eighties, the first laptops were presented and available, building up the basics for the first really mobile applications. The Boeing company were the first using an ARS to assemble cables in the early nineties [36] while at the same time the first smart phone was released by IBM. Furthermore, in 1993 the GPS got operational capability and served as a position measuring system for electronic devices like ARS. With these steps in development, the main fundamentals for todays ARS were set. After this period of time the technical basics became more efficient, more accurate, light-weight and the range of applications increased rapidly. Section 3.2 will show that there are a lot of possibilities to estimate the pose of an user and to estimate the orientation of its view. The section gives a short overview of actual hardware components. To obtain an idea of actual overall systems, study the listing of systems given in 3.3.2.

3.2 Technology of AR Systems

AR has a widespread technological foundation. ARS consist of several different technologies and sensors, e.g. hardware components to capture the circumstances of the real environment and software components to fuse the real and virtual data. The result is a new view of the real environment enhanced with data which seems to be originally captured in this way. To realize this, an AR device has to be able to measure its position as exact as necessary to display the virtual objects in their correct position and orientation. In the best case the user can interact with the environment. For the estimation of position and orientation, the visualization of the available data and the possibilities to interact, an ARS has to unite diverse technologies. The following section gives a short overview and explanation of the different technical components of ARS. The figure 3.2 shows the fundamental architecture of an ARS given by [22]. The authors differentiate between interaction devices and techniques, presentation and authoring and on a more technological level between tracking and registration, display technologies and rendering. I will therefore concentrate on tracking and registration, mathematical background, display technology, rendering techniques and data and interaction techniques. Immersion as an overall characteristic will close this section.

Tracking and Registration An estimation of the pose (position and orientation) usually has to compute six DOFs. The position of a device can be described with three DOFs. Using a *Cartesian* Coordinate System, a tuple (x,y,z) can be used to define an unique position. To estimate such a position there are point-based position measurement units available, e.g. based on the *Global Positioning System* (GPS), or triangulation-based systems measuring several points in the environment

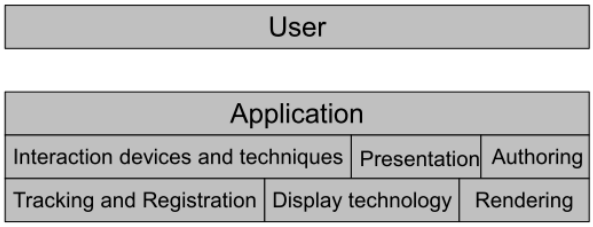

Figure 3.2: Building blocks for AR on the base of [22].

and estimate the position via a spatial triangulation technique. For the knowledge of known points in the environment there are data bases or description grammars necessary. This has to be combined with a technology to detect the points. This approach is called marker based as it searches marked points in the environment. Its main disadvantage is the fact that the environment has to be prepared in advance. For the detection the points can be marked with identifiable sensors (optical like cameras, infrared or by a black & white pattern, radio techniques etc.) to simplify the search. See [60] for a very good overview and a listing of characteristics and drawbacks of possible sensor technologies.
The characteristics that define the overall performance of an ARS [202] [184] are accuracy, latency, jitter, robustness, range of operation, update rate, occlusion behavior and power consumption. Furthermore, there are two types of sensor units, *outside-in*, where sensors track emitters on the user and *inside-out*, where measurement units are attached to the user. Haptics (force, heat, vibration) are for short range AR very important. In the case of a wide range augmentation system, the objects do not need to be touchable[202]. Acoustic tracking based on time of flight measurements of e.g. ultrasonic signals, is another possibility as it measures the distance between receiver and emitter. To date, no ideal overall solution exists. Each application has to consider trade-offs [184]. For outdoor and real-time ARS, inside-out sensors like combinations of gyroscopes, accelerometers and vision-based offer a very good solution. The measurements are linear accelerometers or angular velocity and their integration over time using a very high update rate. The drawback of this technique is a drift in the values over time the algorithms have to deal with [184].
In unprepared environments a markerless approach is fundamental. This kind of algorithms try to identify typical objects that are given by a description or learned by the ARS in advance. By using a point-based position measurement, the orientation can be estimated simultaneous. An orientation can be described by the angles of rotation axes. In body-centered systems the orientation tuple can be defined by *pan/heading, tilt/nick* and *roll*. In contrast to this, a GPS-receiver can not offer orientation information in stationary mode. For this purpose other kinds of sensors are required (inertial with gyroscopes and accelerators, electronic compasses etc.). The first-time pose estimation concerning the real environment is called *Registration* while the continuous observation of the pose is called *Tracking* and is much less time consuming.

Mathematical background To understand the mathematical background we examine the image pipeline. Inherently the three dimensional environment fits to the laws of perspective and space. An optical sensor captures the image by projecting the environment to a two-dimensional plane (image sensor) resulting in the reduction of one dimension. To fuse this two-dimensional image with the computer-generated virtual data, internally stored as a three-dimensional model, the process of the projection has to be known. By a reproduction the virtual data will be projected and can be overlayed onto the two-dimensional image. This projection parameters are unique for each camera mockup and are called *intrinsic parameters*. They can be estimated by calibration approaches. For more information to this topic see [79], [161] or [64]. Secondly, there are the *external parameters*, defining the orientation of the ARS, that will be measured iteratively by sensors. The kind and size of the sensors and the effort of the estimation of the parameters affect the size and mobility of the ARS. For a registration with the correct pose there can be different coordinate systems, depending on the number of sensors and the physical mockup. More details are collected in section 2.

Display Technology The display types can be divided into two groups: *Optical-See-Through* (OST) and *Video-See-Through* (VST) technologies. OST have an optical combiner (transparent display) to mix the virtual and real environment. The other one captures the real environment with a video camera and overlays the images with the computer generated ones. As a display, monitors in front of the users eyes are used.

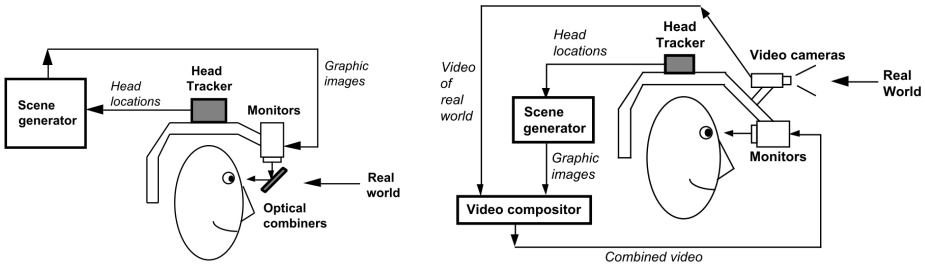

Figure 3.3: Left: OST schemata. Right: VST schemata by [12].

The precise measurement of the heads position and therefore a high accuracy in the registration of the contents mainly defines the performance of these systems. Most systems offer only a FOV of around 35 degree (humans have 55 degree in horizontal range with static eyes), leading to a lack of full immersion. Other problems have been e.g. the uncomfortable wearing on the head with cables, batteries and heavy-weighted components [202]. Figure 3.3 shows on the left hand a typical mockup for an OST device and on the right hand for a VST device. A summary of technological, perceptual and ergonomic issues of AR displays can be found in [150]. Characteristics are latency, update rate, resolution, FOV, optical distortion, brightness, contrast and weight. Types of displays are head-mounted, hand-held and projection-based displays where the last one uses VST technologies and video

projectors to project directly on the surface of physical objects, see [168] for more details [184].

Rendering techniques and Data management To adapt the virtual data perfectly to the conditions of the real environment (light, reflexions or daytime), VR Systems usually use 3D computer graphics which cover a very wide field. There is a PhD thesis which explicitly deals with this topic, see [62] for a detailed and mathematical background. The data an ARS has to display can be provided in different ways. The data can be allocated by databases, object and environment descriptions in form of grammars or complete maps or charts. Nowadays streaming the data over networks is the mostly used approach. Often the problem is the mass of data which has to be well prepared in advance. For the topic of information filtering for mobile AR see [101] or [187]. Especially in the case of ubiquitous computing, for which the devices should be integrable into the daily life seamlessly, [185] can give a good overview for the vision of ARS.

Immersion The immersion is an overall characteristic illustrating the value of accuracy and real-time performance. Immersion describes the degree of realism. The highest realism can be imagined with display technologies which are close to the eyes and an error-free registration algorithm combined with perfectly illuminated virtual objects with no latency. Until now, the highest degree is not reached and every ARS has some lack of immersion due to a smaller FOV or latency errors in the renderings. The errors and insufficiencies of all internal components sum up in an ARS and the worst case can cause the so called cybersickness [16], which describes physical stress and indisposition of the user.

3.3 Thematic Prior Work

The goals for this work are clearly defined in section 1.2. There is no solution yet, which fits all requirements we need for this maritime application. Due to the development of an overall prototype there are multiple challenges. First of all, the demands on the tracking accuracy are very high, caused by the magnification factor in combination with a very small FOV of approximately 10 degrees. Secondly, the system will be an outdoor ARS with the condition to augment wide range data with distances that range from 500 m up to 5 km. Thirdly, outdoor tracking devices are based on markerless tracking, dealing with object detection and pattern recognition. Most applications have to learn the main context of the scene with a large database of example images, which means a lot of effort in advance. This prototype will use the object descriptions given by sea charts and therefore there will be no learning phase. Fourthly, to effectively use the chart data, this approach extends standardized sea charts by a hierarchical ontology which allows a preselection and reduction of the number of possible objects. Influenced by these four points, the following prior work selection is divided into the parts: general AR applications, outdoor AR systems, integration of visual sensor technologies and the usage of maps by tracking approaches.

3.3.1 Existing AR Systems and Software Platforms

To give a starting point, this section of prior work is focused on general applications. Applications of AR range from superposition of virtual objects in TV and film productions [199] to online support for industrial service maintenance [108]. An impression of the wide field of possible applications was given very early by Azuma in 1997 [12]. The author selected the sectors medical, manufacturing and repair, annotation and visualization, robot path planning, entertainment and military aircraft. Until now this field increased much more, so that this section can not handle all of them in detail.
A domain with high effort in the generation of new approaches is the military one. In the beginning the military used AR to provide *Head-Up-Displays* (HUD) for pilots in aircrafts in the topic of navigation and *Situation Awareness* (SA). Interesting information were e.g. radar images, night vision, data concerning the own and the foes position and weather data. This development was later adapted to automobiles [30] and today it is available in many civil cars. The advancement of the HUD has been the *Helmet-Mounted Display* (HMD), to support the information display at every moment, independent of the user movements. A great 25 years history of this topic can be found in [166], and further details in [134] or [167]. Simulation and training is the second application in the military [100], [122]. To adapt these technologies into real combat action, they should have reached an everyday robustness and reliability, because there is the responsibility to not risk life. Therefore, until now, the ARS found a very accepted broadness in simulation, training and military education [143], [100], but there is still a large interest in innovative developments to bring this technology into the everyday military life.
In contrast to the military field, the civil and industry sector evolved more slightly. But even in the civil sector, a number of AR applications are well-established and accompany us in everyday's life. In television, AR technology supports the visualizations in sports, e.g. by displaying the accurate finishing line or helps with effects for the film industry. As an example, actors can be tracked by motion capture technologies and by this motion data virtual characters are animated. Also in the advertisement field, the customers can get several visualizations of the product by marker technologies on the packaging or can view whole videos with their smart phone by filming an advertisement in a journal. Especially the market of games had a big benefit of AR technologies, as *ArQuake* [151], or the *Kinect Software Development Kit* (SDK) show. ARQuake is a user centered game which allows user to move in a virtual game which can be played outdoors. The Kinect allows gamers to move their arms and legs for their virtual characters. The technology of the Kinect SDK is given by [194]. The field of games is huge and particularly increased by mobile phones and their wide distribution. These applications are not high professional solutions, but they pave the way for more reliable applications like logistics [192] or medical support [80, 205, 53]. For the manufacturing case, Regenbrecht et.al. [169] and [108] give a good overview of projects in the automotive and aerospace area. The next main application fields are navigation and city exploration, education and data examination. Examples for city explorer are *Archeoguide* [219], *TOWNWEAR* [179], [163] and *ARVino* [107]. Also the navigation software for cars improved by

AR technologies, as *Wikitude* [2] shows by adding AR to video navigation. In education AR has been used for geometry education [103] or mathematics [104] and for the visualization of data. An example is [175], which explains the visualization of subsurface features. Another interesting sector for this purpose is the maritime one. The authors of [48] give a good overview of VR and AR in the maritime industry. Others try to enhance chart data with panoramas [154], for building the 3D Chart-of-the-Future Display. Beside the publications mentioned, there are much more applications like the detection of floating objects on the water surface like mines [27] or [115], where the authors try to detect anomalies in ship routes. Beside the wide range of mobile hardware, the software development has become a business area. There are diverse platforms (*iOS* SDK[95], *Android* SDK [9], *Palm* WebOS [88], *Windows* Mobile 7 [148], *PlayStation* 3 [153] and many more) and toolkits (*Studierstube* [69], *ARToolkit* [212], *Tinmith* [113] and many more) to develop AR software applications [52] available. Today there exists a wide range of tools for the authoring and publishing of tools and standards. For detailed information about the tools and development kits see [32].

3.3.2 Outdoor Markerless Tracking

As a result of well performing indoor ARS, there has been high effort in the development of outdoor applications in recent years. There are some important differences to indoor applications as stated in [14]. All resources are limited. As the user has to carry everything, the constraint of a maximum of weight influences the power supply, the number of batteries and therefore the power of the computation and sensor units. Beside the conditions concerning the capacities, the environment is not prepared and not controllable.
The light variations have a wide range, as they differ from bright sunlight to dark night. Most display devices can not deal with these conditions and can not display this range of values. These differences show, that indoor solutions normally can not be applied to outdoor environments. In [67] a discussion is given, concerning the possibilities to receive robust outdoor augmentation of sport scenes. Outside-in tracking approaches can serve as an example. An instance of them is explained in [132]. Here, a marker tracking with high accuracy with 0.5 degree of head pose with infrared light-emitting diodes (LEDs) is used, attached to the users head to track the position and orientation. This solution is not applicable to outdoor environments as they need a prepared environment.
For outdoor ARS there exist several possible solutions in literature. First of all, the main question is, how to estimate the pose in an unprepared environment. Either there is a description of the scene, given by some kind of map or there is none, and the measurement of pose has to be done in a self-referred way by adequate sensor technologies. There have been diverse approaches, which have to be reviewed in this context. Using sensor technologies without visual tracking is an often used approach. The paper of [163] explains a prototype for a hand-held ARS that uses a digital database of the topography of Great Britain with natural and man-made features. The pose estimation is done by a GPS- and a digital compass sensor. It shows orientation errors of $\pm$ 0.5 degrees and translation errors of $\pm$ 2 m. A similar

system is shown by [11]. The hybrid tracker combines rate gyros with a digital compass and a tilt orientation sensor in a near real-time system. The authors state that "noise in the compass output makes the labels jump around by 0.5 degrees or more". [181] showed a hand-held ARS for underground infrastructure visualization. The estimation of the pose is done by a GPS sensor and an IMU which includes a digital compass. Problems occur, if the user moves in street canyons, because of the shadowing effects on the GPS signal by high buildings. IMUs are sensor systems with adequate measurement possibilities for pose estimation and they utilize a digital compass. This encourages developers to use the included digital compass 2.1.1.3.1. Many more examples could be given [96], but due to the ferrous materials on a ship, digital magnetometers are not applicable to estimate the orientation. For the same application of underground infrastructure visualization there is a similar approach by Roberts, Evans et. al. [175]. They used the GPS sensor with an INS with a digital compass. Other solutions of orientation tracking are e.g. [107]. The authors use a 'turn-table' platform for the computation unit and a magnetic orientation sensor to visualize *Viticulture* GIS Data. The 'turn-table' supports the orientation sensor if distortions happen. It would be possible to use maps of the distortion to calibrate the magnetic sensors, but in the case of other electronic equipment in the ARS, the real distortions are hard to predict.
The procedure of pose estimation with the generation of a map is covered by the topic *Simultaneous Localization and Mapping* (SLAM) which is often used for the navigation of independent robots.
In this field, beside additional sensor technologies like odometers and laser or radar technologies, the core of approaches mainly use the hybrid technologies of GPS, inertial and visual estimations. As an example for an obvious indoor application, [38] presents a 3D motion and structure estimation by using inertial sensors and a *Kalman Filter* (KF) approach. The method fuses data from head- mounted cameras and head-mounted inertial sensors with *Extended Kalman Filters* (EKF). The first estimates the motion of the user's head and the second estimates the 3D locations of points in the scene. This works best with stereo vision, but only for an indoor scene.
The field of SLAM is very large and many developers have improved the algorithms for many years. Todays publications mainly improve the robustness like [25], which uses optical flow to reduce the needs of 3D correspondences in a tracking scene. Recently, the methods of SLAM have received great interest in the field of AR, as they become stable enough to deal with the requirements of changing lights and dynamic environments. Furthermore, real-time performance is given, as [47] shows. The approach performs an online creation of a sparse but persistent map of natural landmarks within a probabilistic framework. The topic of using maps for a robust pose estimation is handled in more detail in section 3.3.4. The last paragraphs have shown multiple possibilities of sensor-based position estimation techniques. Due to the low accuracies of IMUs, the inaccessible but precise INS and the problems with magnetic compasses, many researchers add a vision-based approach to stabilize the measurements.
As the publications show, the estimation of the position can be done quite accurately with a GPS sensor, if the ARS is used outside. In this context, the GPS sensor is a

good choice for the outdoor usage. In the case of the usage within a ship, e.g. on the bridge, another solution is needed. The more challenging part is the estimation of the accurate orientation, as the IMUs have standard deviations of 0.2 degree per second and need a global reference.

3.3.3 Integration of Visual Sensor Technologies

Very early, in 1999, [220] and [231] state, that IMUs as a stand alone are too inaccurate for a robust pose estimation, same for the optical flow approach, but they also showed that the hybride approach that uses inertial and visual tracking gets the best results. In 2003 Caarls, Jonker and Persa [57] proposed a sensor fusion approach for outdoor ARS. They combined inertial sensors with a compass, a GPS unit and a marker tracking approach. They reached high accuracy only in the near environment. In [196] a comparison of different position technologies, like GPS, WiFi, vision based positioning and dead reckoning is presented and identifies the integrated vision based positioning as necessary. [232] combines low frequency stability of vision sensors with the high frequency tracking of gyroscope sensors with an EKF. The results show much more stability and robustness compared to the sensors alone. To integrate vision into the estimations, there are two possibilities: matching of the captured images with a 2D representation of the scene (reference images) or with a 3D representation (a 3D model).
In the case of 2D matchings, a broadly used approach is the construction of a panoramic image model of the actual environment. [198] shows a maximal error of the tracking in panoramic images of 0.9 degree. Similar results give [40]. [186] proposed a pose estimation by a combination of a GPS sensor, an inertial measurement unit and a visual panorama tracker to compensate the errors of the digital compass and the drift of the accelerometers. The panoramas are mapped on the fly and a markerless tracking approach tracks the orientation. For the accuracy of the visual tracking alone, with the camera fixed on a mount, they show a standard deviation up to 0.3 degree. Unfortunately, they did not show the detailed values for the overall performance with the Kalman tracking and the graphs are not precise enough to show if the tracking accuracy lays under 1.0 degree. Furthermore, the amount of features will not be found on a sea surface without coastal structures in the background. But even in bad conditions, this approach can give good results as [234] presents. Under these conditions of a heavily cluttered environment, the mean angular localization error is less than 3.43 degree, which the authors compare as equal with digital compass measurements. Panoramic image matching can be used with an offline phase where the images of the environment are captured and aligned or the panorama building can be done on-the-fly. The panorama can also be used as a map for a photo-realistic virtual space [6]. The same technology is used by the *Google* Inc. and their *Google Maps* application. If there is no model of the scene and a pure image-based approach should be used, the panoramic image matching is a good choice.
For the construction of correspondences between the images there exist several approaches. One approach is to model the underlying geometry by a homography and the constraints of the epipolar geometry. Bayesian Maximum A Posteriori

probability (MAP) estimation can vote the correspondences [34] or feature-based approaches which have own similarity measurements. Besides the main contribution of their work, they give a good overview on multi-view tracking approaches. Features are local descriptions of the observed image parts. Widely known feature descriptions are SIFT [124], MSER [128], FERNS [116], GLOH [135] and SURF [18]. They can be searched by known image parts (given by a feature data base) or they can be used to build an environment map with appearances and 3D positions. In [35] the authors explain how a pose estimation within a 3D map is accurately possible using a reconstruction of the scene with matched SIFT features and key frames. The high precision shows the need of a feature or image database. For the map building case, there are several publications in the SLAM field, one example of wider use could be the wide area localization on mobile phones, as presented in [188] and [183]. The approach uses sparse 3D reconstruction by natural features for a full 6 DOF pose estimation. [214] showed, that SURF features can handle long periods of time of almost a year where seasonal changes alter the appearance of the environment. Robust approaches in man-made structures are also presented in [108] and [170]. Another possibility is the usage of edge and line detection in combination with an off-line learning phase [82] or [237] with the *Video Compass*. They use the alignment of structures in man-made environments, like principal orthogonal directions and vanishing points. These approaches can be very robust to occlusions, noise and illumination changes. If the feature database is not built on-the-fly and there is no data base to use, the features can be learned in a training phase. This holds also for segmentations, like in [206], or other objects description techniques. Publications that refer to this topic are [172] and [173].
A step forward is the usage of differentiated objects instead of small, independent features. In [8] the authors explain an *AdaBoost*-learning procedure, which votes features and its poses belonging to an object. The best features to detect the object of interest get the maximum voting. Unfortunately, its not suitable for the varying conditions in outdoor environments as the features would have to be given in all possible appearances. Also fuzzy logic was used to detect objects [157] or for scene labeling [87], [145], [142]. The problem with these approaches is a huge data base that has to be built in advance. The data base is used to provide all possible features for the learning phase. The next step is to build a second data base with the combination of interesting objects and the learned classifier. In the context of this work, a learning phase should be avoided, because the application shall work in very different environments without preparation. An interesting paper is given in [5], where *Monte Carlo Particle Filter*s are used to predict head motion. They show a precision of 0.02 degree. The precision depends on the number of features and the slowness of the movements, which can be a problem in this context because of fast user movements, but the approach is interesting. Another solution could be the usage of a 3D model of the environment. [83] shows that knowing the shape or geometry of the objects to track in an outdoor environment, allows a significant improvement in the tracking performance by the usage of affine transformations and surface feature additions. Beside this, they use a laser range finder and near-field AR. If no model is given, the point clouds of *structure by motion* approaches could be used. The challenge of generating higher level models to model real object

shapes is processed by [215] or [21]. The generation of higher level objects is often an approximation with standard geometry shapes, see also [41]. In [68] a knowledge base of the geometry of man made structures guides the modeling process. The authors use a semantical net which simplifies the definition of 3D primitives in the depth maps. This leads to the fact, that these approaches only hold for man-made environments that provide geometrical structures. The challenge of this work is the maritime environment and the detection of navigational signs. These signs include vegetation groups, embouchures and visual conspicuous hills, which do not fit in standard shapes. There exist further beside the already mentioned possibilities, for examples see [159]. The approach is called *visual servoing* and uses solely the visual information without any additional sensor. The orientation measurements by homography and displacement estimations (M-estimation) give good results, but as a disadvantage, the approach needs an initial pose. The last point to state here is the case, where the 3D model of the environment is known. See [193] for a good explanation of automatic model building. A good overview of pose estimation within known 3D geometry gives [125]. Normally, a manually built 3D model of a natural environment needs a huge effort, that should be avoided. Another possibility is to construct an environment model on-the-fly, see in terms of visual grammars [66].

3.3.4 Usage of Maps

There are a lot of approaches that try to use given data, for example known maps. They can be given as paper charts, digital maps or *Geographic Information System* (GIS) charts. [171] states that related work only works with small data sets and that only a few authors have coped with selecting appropriate data for display. The data for their approach was collected very costly by a 3D city model, GIS data and general maps of the city. [24] combined GIS with off-line AR techniques for realistic panoramic video frames. Besides GIS, if there is an elevation map, the horizon silhouettes can be exploited [139], [43], [19]. These approaches can reach heading accuracies of $\pm$ 0.2 degrees, but in this context, the sea surface will not always provide definable silhouettes at the horizon. To use the charts the information has to be available in a digital form. The digitalization of paper maps started very early [203] with weather charts or the recognition of geographical features [49] or the symbols of a chart [178]. Others try to overlay a digital GIS map onto an aerial photo [211]. That AR is an interesting topic for cartography is stated in [184] and presents examples of how AR can be applied. In [225] the future of AR and GIS is discussed as a support for cartographic updating. In the maritime sector, there exist first intentions to integrate sea charts. *NOAA Nautical Charts* can now be overlayed onto Google Maps for a web-based usage and there are first publications on the usage of sea charts for GIS based applications. Others deal with the data model extension for a seamless fusion of land and sea charts [213]. [10] expands the usefulness of NOAA ENCs via the creation of web-accessible GIS, but up to the results of this research, there is no ARS yet that uses neither sea charts nor 3D sea charts as the prototype explained in this work does. Maps or environment models can provide huge semantical knowledge, which can help to navigate. This overview of AR and ARS refers as base knowledge in the context of this work. The specialized

topic and the arising problems of maritime ARS are discussed in the following.

Chapter 4

Maritime Augmented Reality

Contents

4.1 Overview

To get an overview of the system and the components the figure 4.1 serves as an illustration of the resulting system. Starting at the upper left point, the data base delivers the interesting data that should be displayed. In the case of a maritime application, the data base is a sea chart system. The classical sea chart is a 2D chart which is displayed in a 2D top view perspective. As we want to build an augmentation scene, there is the need for a 3D representation of the 2D classical definition. The ECDIS system provides a new 3D module, which is able to render the 2D sea chart as a 3D virtual scene. This fully rendered scene is not constructed

to allow video streams to be displayed in parallel, but it is used in a 3D egocentric perspective. For an augmentation, the scene has to be structured and the pose of the virtual camera in relation to the video streams has to be calculated. To reach this, a sensor system is necessary which is connected to a camera system and a displaying device.

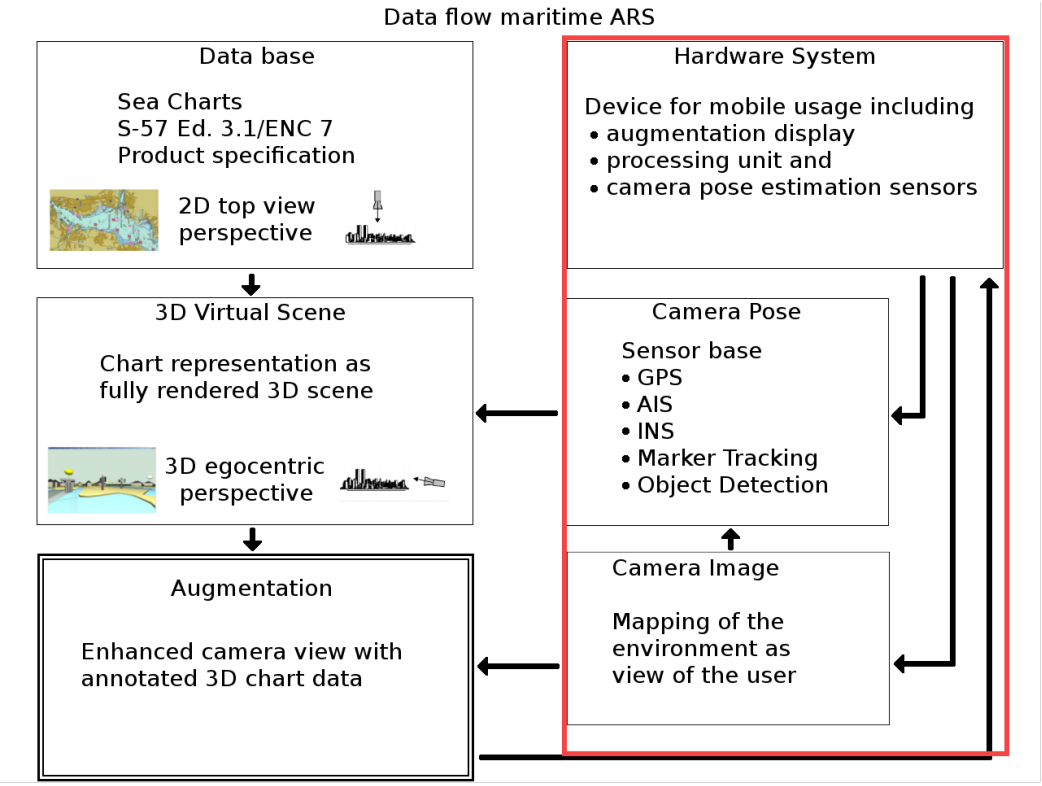

Figure 4.1: Overview of a maritime Augmented Reality System.

This chapter will explain the necessary developments to build a device like mentioned. The first step is the design of a hardware system, which can deal with the made demands. The hardware system presented here is patent-registered. The system with it's sensors is presented and evaluations are shown to characterize the performance. The second section refers to the used data base, here the sea charts. The classical usage of the 3D virtual scene will be shown in the hardware section and it will be seen, that the scene has to be fitted to the requirements of an augmentation scene. The section will analyze the sea chart data structure and a new developed approach to build an standard-conform object-to-object topology will be explained. With the help of this topology, the scene can be interactively utilized by the user and it gives a simplified access to the data for the object detection process. After the sea charts have been adapted to the requirements, the hardware system can be improved by substituting the marker detection approach with an object detection approach. This allows the user to use the device outside, in free movements

and without calibration to the environment. The object detection approach is a probabilistic one, which uses the a priori information in the sea charts to build up Bayesian networks which help to set probabilistic beliefs to classify the objects seen by the camera. To sum up, the important steps of the development are the design of the basic hardware and software device, the improvement of the sea charts to be usable in an augmentation scene and the further development of the device to overcome the drawbacks of the basic system.

4.2 Component Analysis and Processing

This section explains the possible components of a hand-held system for the augmentation of sea chart objects. The special requirements are given in the introduction 1.2. The binocular should be usable in a free hands mode on a ship's bridge. That means, the device is dependent of the position and bearing orientation of the ship and in addition it is dependent of its own fast rotations done by the user. The user rotations are therefore independent from the ship's orientation changes. These relations will be addressed here as "global" and "local" poses. The global context is the ship and the local refers to the device itself.
The aim is a device which can be used everywhere on the ship's bridge, even when going out into the knocks. The observed objects in the camera view can be overlayed by detailed information and 3D symbols given by the sea charts. The objects are expected at distances between 100 m up to approximately 10 km. This leads to the requirement, that the camera has to have a magnification lens, to be able to display the objects at wide ranges. A 25 mm objective on a 1/3 inch sensor produces a mapping of 60 pixel per degree of the environment in the augmentation camera. To reach pixel accuracy in the augmentation the estimation of the viewing direction has to reach a precision of 0.016 degree. The following sections will discuss the components of the developed device.

4.2.1 Estimation of Local and Global Position

In outdoor scenarios the global position can be given by a common GPS sensor. Unfortunately, the error range of a general GPS sensor ranges between 5-10 meters. To achieve a better accuracy, the usage of a differential GPS sensor is possible. An application scenario for the augmentation device can be the usage on the ship's bridge. Due to the indoor character of this context, the ferrous material of the ship will lead to weakened GPS signals. It is more practical to use the given DGPS sensor of the ship, that can be delivered by the listening to the ship's AIS messages. Unfortunately, these messages have update rates of 3-10 seconds. As the ship moves very dull and often straight-lined, the usage of a Kalman Filter approach with linear movement models is possible to refine the measurements. On the other hand, the local translation of the user w.r.t. the ship's bridge position can be neglected, because the range of translation is very small with respect to the distance to the objects. See appendix A.1.3 for the underlying coordinate systems.

4.2.1.1 Kalman Filtering for Global Trajectory Estimation

A Kalman Filter (KF) is a stochastic model to weight measurements (minimizes the error covariance) by an evaluation of their uncertainty, so that with the help of movement models, new measurements can be predicted. Simultaneously, the predictive values lead to an adaption of the underlying assumption due to the evaluation of the new physical measurements with respect to the predicted values. Due to the kind of process the filter should control, the discrete KF is used for linear and the Extended KF (EKF) is used for non-linear types. A KF is usually divided into three stages: initialization, prediction and correction. The initialization is used to give the first guess of the system's state. The state $x \in \mathbb{R}^n$ is modeled as normally distributed variable $x \sim \mathcal{N}(x, C)$ with C as covariance matrix and can be expressed as

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \tag{4.1}$$

with A as evaluation matrix of the states x_k and x_{k-1} and B as optional control values. Additionally, a measurement $z \in \mathbb{R}^m$ is given as

$$z_k = Hx_k + v_k \tag{4.2}$$

with H as matrix that relates the predicted state to the measured one. Note, that w and v are process (Q) and measurement noise (R) models that have to be independent, normal distributed around zero and of a white noise character. That means $p(w) \sim \mathcal{N}(0, Q)$ and $p(v) \sim \mathcal{N}(0, R)$. The error covariance for the prediction is then

$$P_k^{'} = AP_{k-1}A^T + Q. \tag{4.3}$$

The result $\widehat{x}_k$ can be interpreted as a posteriori knowledge given by the history of measurements z_k and refers to the first moment of the distribution $E[x_k] = \widehat{x}_k$. Respectively, the predicted error covariance P_k refers to the second moment $P_k = E\left[(x_k - \widehat{x}_k)(x_k - \widehat{x}_k)^T\right]$. The estimated error covariance is then the variance of the distribution and can be modeled as

$$prob(x_k \mid z_k) \sim \mathcal{N}\left(E[x_k], E\left[(x_k - \widehat{x}_k)(x_k - \widehat{x}_k)^T\right]\right). \tag{4.4}$$

After the prediction and the measurement of actual process values, the measurement update is necessary to correct the made assumptions. For the a posteriori state holds

$$\widehat{x}_k = \widehat{x}_k^{'} + K_k\left(z_k - H\widehat{x}_k^{'}\right) \text{ and } P_k = (I - K_kH)P_k^{'} \tag{4.5}$$

with I as identity, $\widehat{x}^{'}$ as a priori of the state, K is the so-called *Kalman gain* matrix, estimated by

$$K_k = P_k^{'}H^T\left(HP_k^{'}H^T + R\right)^{-1}. \tag{4.6}$$

With these equations all three phases of the KF are described [227], [130], [228]. The EKF uses partial derivatives to linearize the functions around an actual estimate, but the stages in the process management are the same. Both models deliver an optimal estimation by interpolation of noisy measurements and extrapolation of

predicted measurements [102]. To describe the movements of the ship, two linear KF models were used. The first filter estimates the trajectory of the ship by using the GPS signals, and the second filter smoothes the AIS heading messages which are too rough. Both results are given independently to the pose estimation. The trajectory is estimated with the following model:

$$\mathbf{p}_k = \begin{pmatrix} x_1 \\ x_2 \\ \varphi \\ \bar{v} \\ r \end{pmatrix} \text{ and } f(\mathbf{p}_k) = \begin{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \cdot R(\varphi) \cdot \begin{pmatrix} \bar{v}_k \\ 0 \end{pmatrix} \\ \varphi \cdot \Delta t \cdot r \\ \bar{v} \\ r \end{pmatrix}, \tag{4.7}$$

with x_1, x_2 as position, φ as estimated direction of the movement, $\bar{v}$ as velocity and r as angular rate. The system parameters were set to:

$$P = \begin{pmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.01 \end{pmatrix} \tag{4.8}$$

$$Q = \begin{pmatrix} 10 \cdot \Delta t^2 & 0 & 0 & 0 & 0 \\ 0 & 10 \cdot \Delta t^2 & 0 & 0 & 0 \\ 0 & 0 & 0.7 \cdot \Delta t^2 & 0 & 0 \\ 0 & 0 & 0 & 0.7 \cdot \Delta t^2 & 0 \\ 0 & 0 & 0 & 0 & 0.001 \cdot \Delta t^2 \end{pmatrix} \tag{4.9}$$

and the state transition matrix

$$A = \begin{pmatrix} 1 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & \cos(r\Delta t) & -\sin(r\Delta t) & -\sin(\bar{v}\Delta t) \cdot \Delta t \cdot \bar{v}_1 - \cos(r\Delta t) \cdot \Delta t \cdot \bar{v}_2 \\ 0 & 0 & \sin(r\Delta t) & \cos(r\Delta t) & \cos(r\Delta t) \cdot \Delta t \cdot \bar{v}_1 - \sin(r\Delta t) \cdot \Delta t \cdot \bar{v}_2 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.10}$$

The filter fails, if the motion model does not fit the movements of the ship, which will happen in cases of no movements or fast rotations.

4.2.2 Estimation of Global Orientation

The global orientation of the ship is given by the ship's compass in very high precision. The AR device should be usable in free-hands mode, which does not allow a constant connection the the NMEA bus. Other possibilities to connect to the ship's compass in a cable-free manner are not easy to install. A solution for this problem is to benefit from the prepared data for the AIS messages. By an AIS pilot plug with a WLAN module, the device can get heading information with low effort. Unfortunately, the coding of the heading information is restricted to nine bit, which leads to a maximal resolution of 1.0 degree. Additionally, the values represent the heading in the past. Due to the precision requirements this can only be used as a base for the pose estimation.

4.2.2.1 Kalman Filtering for Global Orientation

To use the AIS messages the delay has to be reduced and the present values have to be predicted. To describe the heading of the ship, the system uses a second KF model:

$$\mathbf{p}_k = \begin{pmatrix} \varphi \\ w \end{pmatrix} \text{ and } f(\mathbf{p}_k) = \begin{pmatrix} \varphi + \Delta t \cdot r \\ r \end{pmatrix}, \tag{4.11}$$

with φ as orientation change and r as angular rate. The system parameters were then given as:

$$P = \begin{pmatrix} 15.0 & 0 \\ 0 & 15.0 \end{pmatrix} \tag{4.12}$$

$$Q = \begin{pmatrix} 0.001 & 0 \\ 0 & 0.1 \end{pmatrix} \tag{4.13}$$

and the state transition matrix

$$A = \begin{pmatrix} 1.0 & \Delta t \\ 0 & 1.0 \end{pmatrix}. \tag{4.14}$$

4.2.3 Estimation of Local Orientation

The position data of the sea charts is encoded in spherical GPS coordinates. We can map the coordinates onto a local Cartesian plane and restrict the local environment to 15 km, which is sufficient to avoid measurement errors caused by the neglected curvature of the earth surface [59]. The local approximating plane is oriented tangential to the earth surface with its surface normal pointing towards the earth center of gravity. The plane also defines the horizon plane with good approximation. While the estimation of the orientation by a 6 DOF marker approach is not precise enough, deduced from the correlations between translation and rotation, the surface normal of the horizon plane can be estimated conveniently and with great precision using an accelerometer sensor that provides the gravity g. It is now easy to attach an accelerometer onto a device to measure the local tilt and roll of the AR device. Thus, the sensor delivers two of the three rotation angles of the device. The rotation is defined by the rotation angle α and the rotation axis $d^{'}$ by:

$$\alpha = \arccos\left(g_c^T * a_{up}\right) \text{ and } d^{'} = g_c \times a_{up}, \text{ with } d = \frac{d^{'}}{|d^{'}|} \tag{4.15}$$

with g_c as surface normal in the CCS and a_{up} as corresponding up-axis. The heading direction can not be recovered from sensor data since it is defined by a rotation around the surface normal axis.

4.2.3.1 Orientation Sensor Analysis

The accelerometer component of an inertial sensor can be used for gravity estimation. The gravity vector is given as direction vector in the local ICS, measured as acceleration in m/s^2. The sensor delivers gravity measurements at 100 Hz, but the

data is quite noisy and susceptible to vibrational errors. To characterize the sensor data, the device was mounted on a calibrated *Pan-Tilt Unit* (PTU) with controlled angular movements.
Errors reached up to 15.0 degree deviation in the beginning of controlled single step movements. Moreover, the noise of a sensor is a problem when aiming at high accuracy orientations. In this characterization the measured Root Mean Square error (RMS) of the sensor output is 0.1365 degree, while the observed non-linearity of the sensor is very small.
To further improve the data, the raw data is filtered by an integration over time with respect to the delay between image processing and the sensors sample rate. Therefore a Gaussian window is used which is centered at the date of the actual camera image. A certain augmentation delay could be observed on the camera pitch, which causes a misalignment of about 1.0 degree during pitch rotation. However the misalignment vanishes fast when holding steady. Because the delay time is small, it does not affect the augmentation. The noise is now reduced to a RMS of 0.025 degree, which is sufficient for such an application.

4.2.3.2 Heading Estimation

While pitch and roll can be determined with sufficient accuracy, the heading can not be estimated in this way. Traditionally, heading information is provided by com-

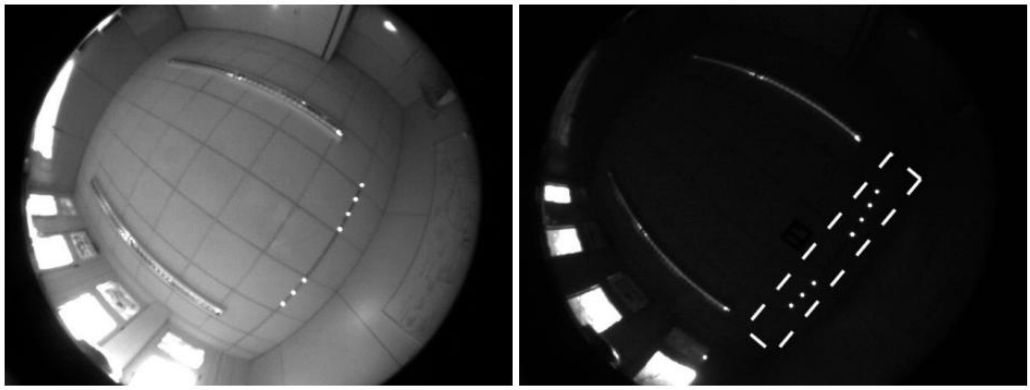

Figure 4.2: Left: Original fish-eye image. Right: Fish-eye image with infrared filter. The marker array is highlighted by the dashed line.

pass data on the ship, but a magnetic compass attached to the electronic binocular will not work with high precision due to the metal construction of the ship. Other sensors or approaches have to be evaluated to estimate a very precise heading. Due to the fact, that the device needs a camera to display the environment, this sensor could be used with Computer Vision techniques to estimate the heading information. For the optical detection of scene context, knowledge of the environment has to be available. If knowledge can be provided by environmental maps, object detection approaches can be applied. If not, a map can be built by learning techniques or artificial objects can be arranged in the near field.

For the application scenario on the ship's bridge a marker-based approach is adequate. To choose a well-fitting estimation technique, a review of the conditions is necessary.
Additionally to the ship's movements, which result in a moving environment, the light conditions are very wide spread. All light variations of the sun are possible, from cloudy sky till bright sunlight. This refers to changes in the light temperature, the light intensity and it refers to very fast changes between them. Beside this, the nautical staff should not be limited in their actions on the bridge, because the marker has to stay in the sight of the detecting camera at any time. To manage the requirements a very robust marker-based approach has to be used.
The approach for this context is explained in the following. Marker-based pose estimation limits the user movements, because the marker always has to be in the FOV of the sensor. To reduce this effect, an image sensor with a fish-eye camera with 190 degree opening angle can be used. Unfortunately, this results in a decrease in resolution of the camera. Such camera with a resolution of 640x480 pixel denotes that only 3.18 pixel correspond to 1.0 degree of the real environment. We require a precision of the marker orientation of 0.016 degree to achieve pixel accuracy in the augmentation camera. To reach this, the detection of the marker within the fish-eye camera has to be with a precision of at least 0.053 pixel. Since only the heading angle is unknown, a 1-dimensional marker is satisfactory. To deal with various light

Figure 4.3: Device on a ship's bridge (at the ceiling right).

changes, an infrared LED array should be used. If the marker is mounted on the ceiling of the ship bridge it can be seen from all positions by the fish-eye camera. Because the heading is measured in the horizon plane of the local coordinate plane, pitch and roll of the AR device will cause perspective distortions, that need to be compensated. This can be achieved by realigning the images with the horizon plane. The used marker consists of seven infrared LEDs, with wide opening angle (flat lenses), see figures 4.2 and 4.3 for image examples.
The marker detection is then performed by a template matching approach to find the lights, followed by an estimation of the real centers of the light sources [229].

To obtain a heading angle in global coordinates the marker line has to be calibrated once with respect to the ship's compass. The marker is additionally equipped with an orientation sensor similar to the one of the ARS to provide the ship movements in the horizontal plane and the compass of the ship provides the global heading of the ship.

4.2.4 Sea Charts

The term sea chart system is a collective term which stands for all possible kinds of displays of sea charts. Components of a sea chart system on a ship's bridge are computational unit, electronic display, an electronic data base, interface for navigational sensors and an interaction scheme. Various tasks have to be fulfilled by a sea chart system.

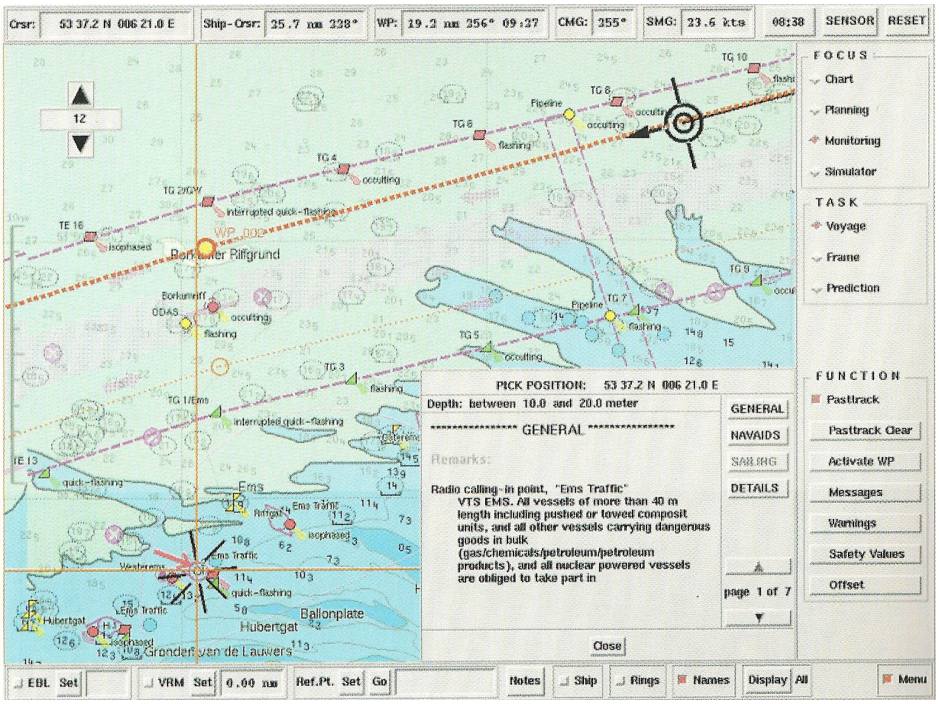

Figure 4.4: A chart with information display is shown [31], originally by SevenCs.

An ECDIS (Electronic Chart Display and Information System) is therefore a comprehensive system, firstly defined in a working paper in 1986 and updated to a performance standard for ECDIS systems [144]. As these systems are allowed to replace the paper charts, the kind of chart data with corresponding display characteristics had to be defined. The ECDIS software used in this context is provided by the SevenCs GmbH, which is a market-leading software company for ECDIS. *Electronic Navigational Charts* (ENC) are defined by the *International Hydrographic Organization* (IHO) in the Transfer Standard for Digital Hydrographic Data, Edition 3.1(IHO-1), Special Publication Number 57. For ECDIS systems and the usage

of nautical charts exists a product specification called ENC/PS to which the data also refers [31]. The figures 4.4 and 4.5 show examples of ECDIS sea charts.

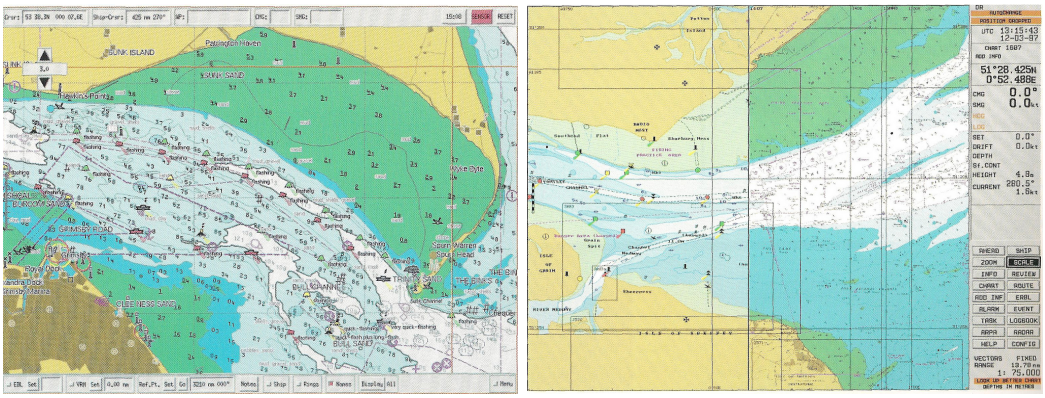

Figure 4.5: The images are screen shots of ECDIS displays. Left: A chart is shown with the display category "all other objects". That means all information beside the security-relevant objects are shown. Right: The image shows a doubled display with vector (left) and raster data (right). These illustrations show the complexity of the display system [31], originally by SevenCs.

4.2.5 Modularization

The application has been tested with different hardware constellations. In figure 4.7 three of them are shown. The image on the left shows a variant which is very closely related to a common binocular which performs with a laptop as base station in the background. The communication between both is realized via WLAN. The image in the middle shows a very small hand-held which has no sufficient rendering support. For this sensor combination a Remote Cube Mapping approach was implemented which allows the device to handle the 3D scene. The last image on the right shows a variation with a touch screen display. With this solution the user is able to interact with the objects, e.g. by pointing to them.
The implementation of the software has been done in a modular way to be able to handle many hardware variations. The given system is visualized in figure 4.6. It becomes clear that each sensor can be exchanged to secure the best combinations for the required system. The architecture shows a differentiation between the display hardware, the rendering system and the pose estimation. The rendering system is dependent on the underlying graphics hardware, which can be very limited on hand-held systems. For this purpose a Cube Mapping approach is available that renders the scene in a panoramic view in advance. The pose estimation can be divided into the different degrees of freedom. This shows the different combination possibilities for this device. While the roll and tilt estimators are only variations within one sensor technique, the heading estimation can be done in different ways. Beside the mentioned marker-tracking other possibilities are the estimation by using the angular rates of a gyroscope and object detection approaches. The last topic is subject to the following sections. A perspective camera for augmentation and a

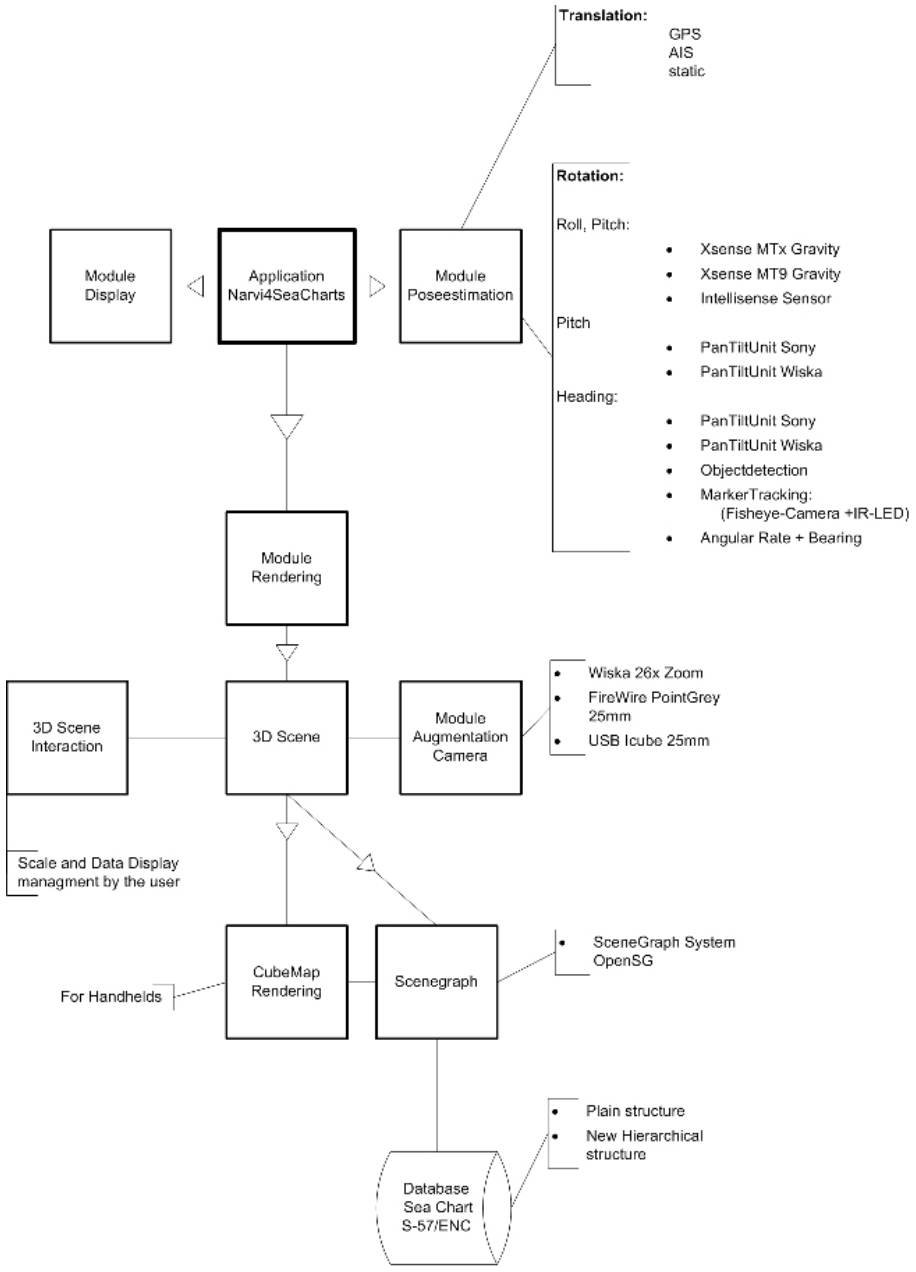

Figure 4.6: Schematic system architecture.

Figure 4.7: Examples of different hardware configurations for the AR device.

fish-eye camera for the detection of the heading marker is used. Additionally, the system is equipped with an accelerometer and for testing purposes a GPS sensor. The monocular images of the real environment and the augmented scene were represented in a binocular VST- display. The system performs with 25 frames per second on a standard laptop and appears very similar to an analogue binocular with its dimensions of 100 mm (w) x 200 mm (l) x 120 mm (h) and a weight of 600 gram.

4.2.6 System Evaluation

The system setup allows the multiple sensors to provide independent measurements. To give an overview of the used algorithms in this system, see [84] for a wide overview of calibration algorithms for vision and inertial sensor technologies. The front camera used in this context has a resolution of 752x480 pixel WVGA with a 1/3" chip element with a pixel size of 6x6 μm with a global shutter. Experiments on noise measurements show an overall noise value of standard deviation of 6.9 for the red channel, 6.15 for the green channel and 4.47 for the blue channel.

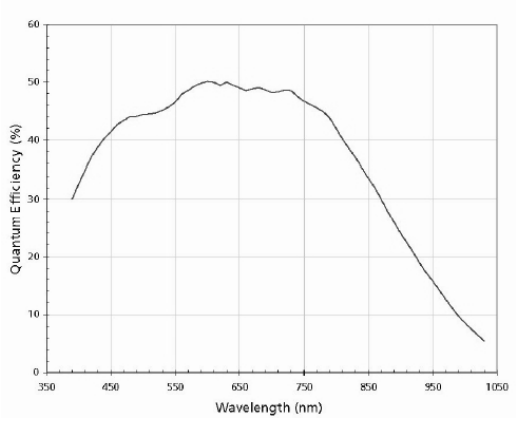

Figure 4.8: Technical characteristics of the monochrome camera, quantum efficiency due to different wavelength

The marker camera is a black and white camera of the same type. The sensibility to different wavelengths is given in figure 4.8[1], which shows the sensitivity to in-

[1] www.net-gmbh.com

frared wavelengths (above 680 nm). Experiments of delay estimations show, that between image capturing and display 144.8 ms go by, which refers to 4.7 images at a frames per second rate of 30. The used lens is a 25mm objective with a geometrical distortion of 0.09% at 0.5 m distances.

4.2.6.1 Position Estimation

4.2.6.1.1 By AIS/GPS The GPS sensor of the ship delivers positions with high precision. To be able to allow the ARS a free movement, it is not connected to the ship's data bus.

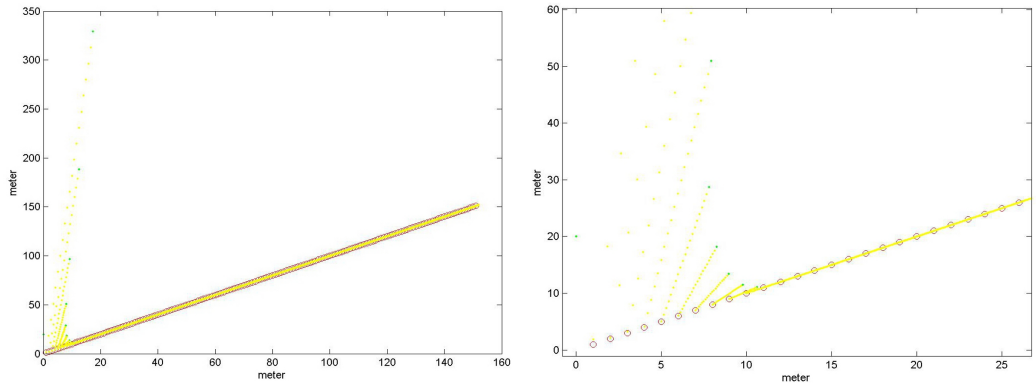

Figure 4.9: Illustration of the behavior of the used KF. Position changes of the track: linear movements. Red dots refer to the measured data, yellow dots show the predictions of the next 10 positions, green dots refer to the predicted positions. See appendix A.9 and A.10 for a bigger plot. Right: Zoom into the left image.

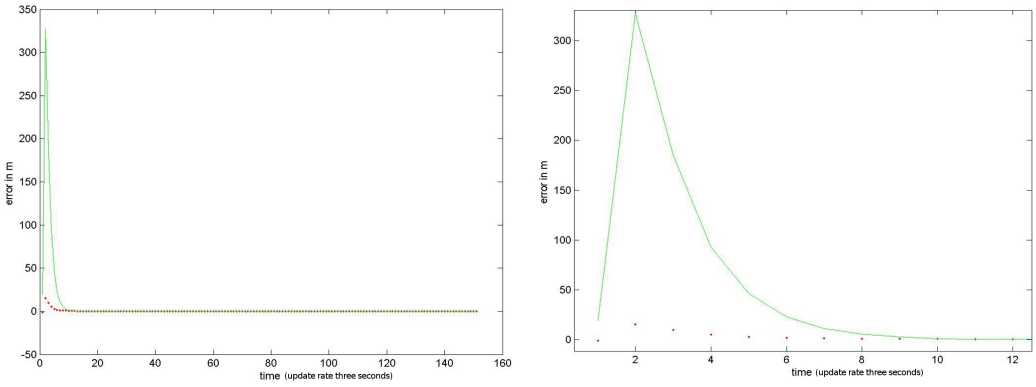

Figure 4.10: Illustration of the behavior of the used KF for linear movement. Error between the predicted and measured values, green line refers to x-positions, red dots are y-positions. See appendix A.11 and A.12 for a bigger plot. Right: Zoom into the left image.

error	Means_{1-8}	Means_{9-14}	Means_{15-100}	RMS_{1-8}	RMS_{9-14}	RMS_{15-100}
x	-40.13	-2.38	0.002	60.08	3.61	0.0009
y	-70.76	-9.93	0.24	102.71	12.46	0.15

Figure 4.11: RMS estimation of the linear movement. After 15 iterations, the error is minimized.

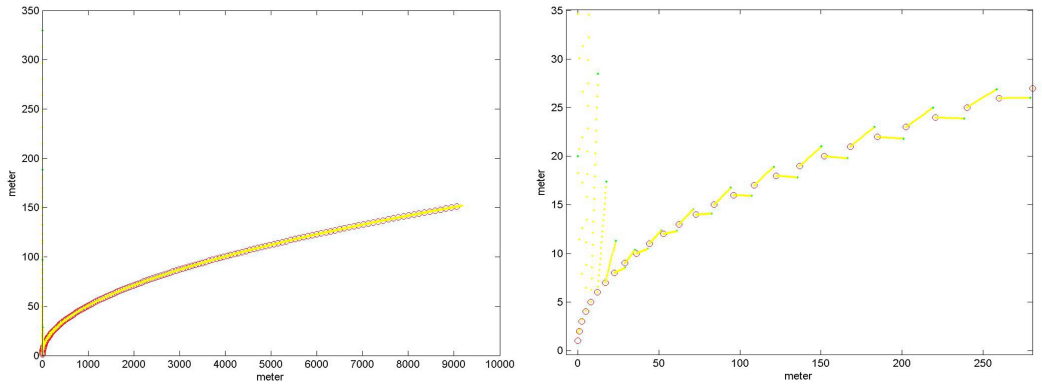

Figure 4.12: Illustration of the behavior of the used KF. Position changes of the track: curved movements. Red dots refer to the measured data, yellow dots show the predictions of the next 10 positions, green dots refer to the predicted positions. See appendix A.13 and A.14 for a bigger plot. Right: Zoom into the left image.

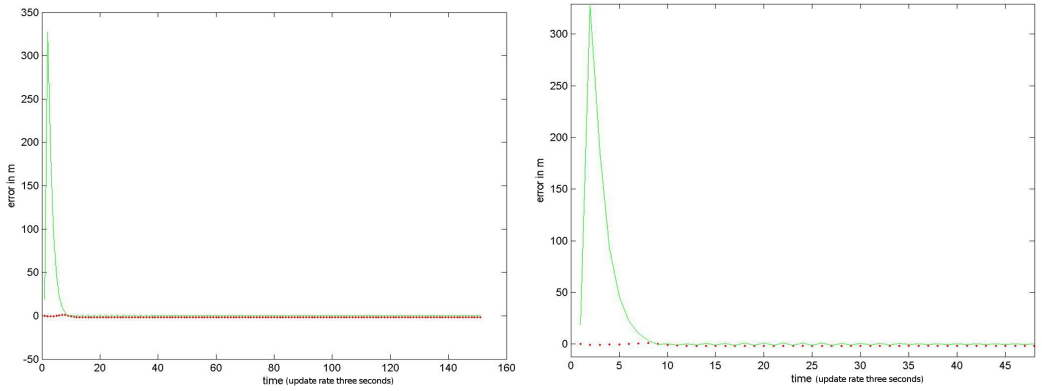

Figure 4.13: Illustration of the behavior of the used KF for curved movement. Error between the predicted and measured values, green line refers to x-positions, red dots are y-positions. See appendix A.15 and A.16 for a bigger plot. Right: Zoom into the left image.

error	Means_{1-12}	Means_{13-100}	RMS_{1-12}	RMS_{13-100}
x	-31.66	3.99	50.76	4.02
y	-51.84	0.62	82.36	0.73

Figure 4.14: RMS estimation of the curved movement. After 60 iterations, the error remains at a small level.

Due to this, the data of the ship's GPS sensor is not available. The solution to recieve information of the ship's position is to listen to the AIS data, which every professional ship has to send. The drawback of this approach is the problem of outdated data, because the sending rate of the AIS data lies between 3-10 seconds. Therefore it is necessary to use the old position data to estimate a velocity model and to use this to predict the actual position and bearing. The simulated behavior of the Kalman Filter mentioned in section 4.2 is shown in the following figures. The parameters are the same as presented in section 4.2.2.1. The figures 4.9, 4.10 and 4.11 show the Kalman Filter in the application of a linear movement. Shown are the track and the error plot of the predicted and measured position. Since the observed targets are far away from the ship, position uncertainties of the ARS due to movement inside the ship's bridge and short term ship motion can be neglected. The figures 4.12, 4.13 and 4.14 show the behavior of the filter when the heading is changing monotonously. The last figures 4.15, 4.16 and 4.17 show a sinus like movement to illustrate heading changes in different directions.

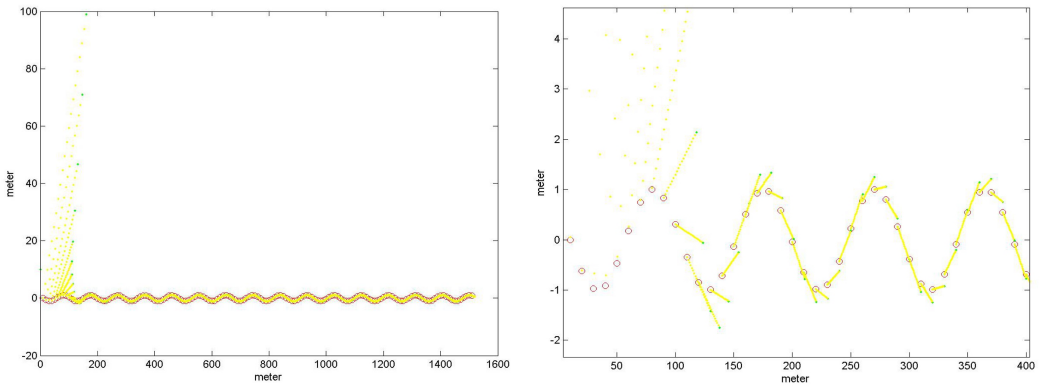

Figure 4.15: Illustration of the behavior of the used KF when moving in sinus-like movements. Red dots refer to the measured data, yellow dots show the predictions of the next 10 positions, green dots refer to the predicted positions. See appendix A.17 and A.18 for a bigger plot. Right: Zoom into the left image.

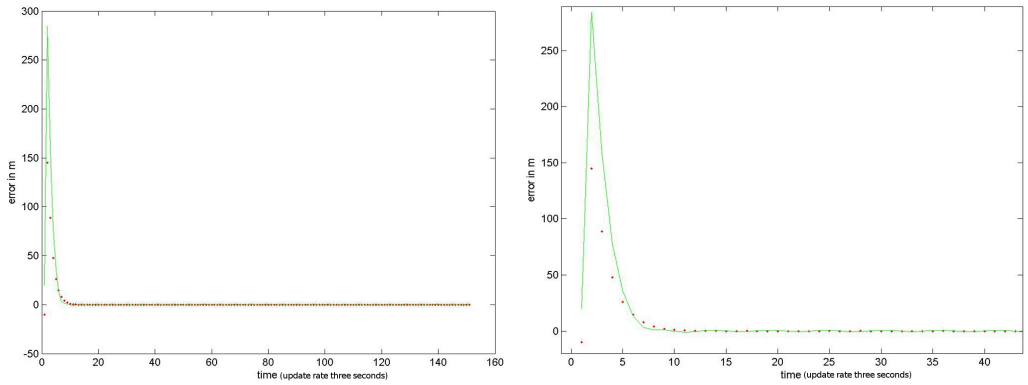

Figure 4.16: Illustration of the behavior of the used KF when moving in sinus-like movements. Error between the predicted and measured values, green line refers to x-positions, red dots are y-positions. See appendix A.19 and A.20 for a bigger plot. Right: Zoom into the left image.

error	Means_{1-16}	Means_{17-59}	Means_{60-150}	RMS_{1-16}	RMS_{17-59}	RMS_{60-150}
x	-45.06	-0.13	-0.13	75.01	0.28	0.21
y	-23.05	0.01	0.003	38.08	0.73	0.44

Figure 4.17: RMS estimation of the sinus-like movement. After 60 iterations, the error converges to a small value.

4.2.6.2 Orientation Estimation

Orientations in world coordinates are estimated by the gravity sensor for the horizon plane, and a heading estimate independent from the ship's compass via the fish-eye camera. The sensor delivers drift-free data that is filtered to reduce noise.

4.2.6.2.1 By AIS/GPS The AIS data also gives information about the heading of the ship. This data is cut to an integer value, so the maximum accuracy is 1.0 degree, which is not enough for this ARS. Furthermore, the data is outdated. To manage the actual heading, a KF is used to predict the values. The figures 4.18, 4.19 and 4.20 show different results of the application of this filter. Due to the slow heading changes of a moving ship, the KF refines the heading values very well.

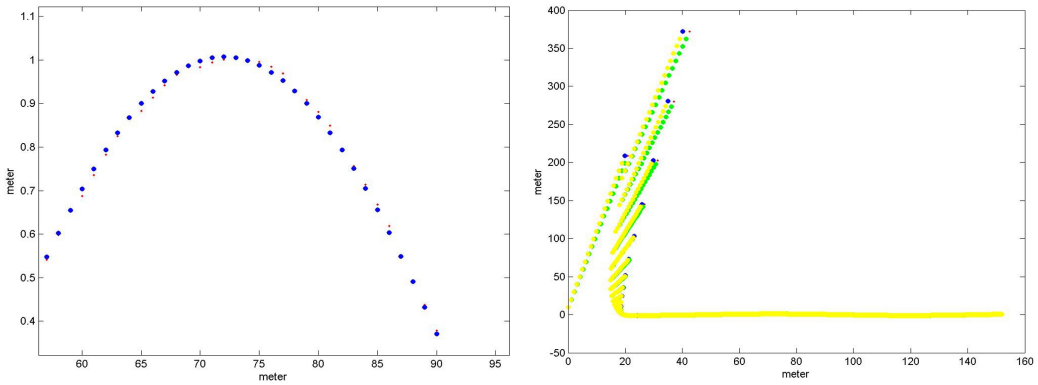

Figure 4.18: Results for the prediction of the heading. Left: Red dots are the ground truth, blue dots the positions if the heading is used as integer value. Right: The difference that occurs by using the rough heading values by the AIS data (yellow) or the predicted and therefore refined ones (green) is shown. See appendix A.21, A.22 for a bigger plot.

4.2.6.2.2 By Inertial and Image Sensors The local orientation can be estimated by different sensors. The IMU sensors will support the estimation with measurements of 2 DOF. For the heading estimation, the usage of the gyroscope in the IMU is possible and an image processing approach can be used. It will be shown, that the demand of precision will lead to the usage of the image processing approach.

4.2.6.2.3 Pitch/Roll Estimation As figure 4.21 shows, the gravity measurements jitter a lot in the raw data. For optimization, a filter is used, which weights the data by a Gaussian function and takes care of the delay in the measurements by using an offset value. This time-dependent Gaussian Mean can decrease the RMS from 0.1365° to 0.025° in the still mode. If the sensor is moved, the raw data plots show the measurements of the sensor which tend to overshoot. The figures 4.22 and 4.23 show the plots and furthermore the behavior of the filtered sensor with different filter parameters.

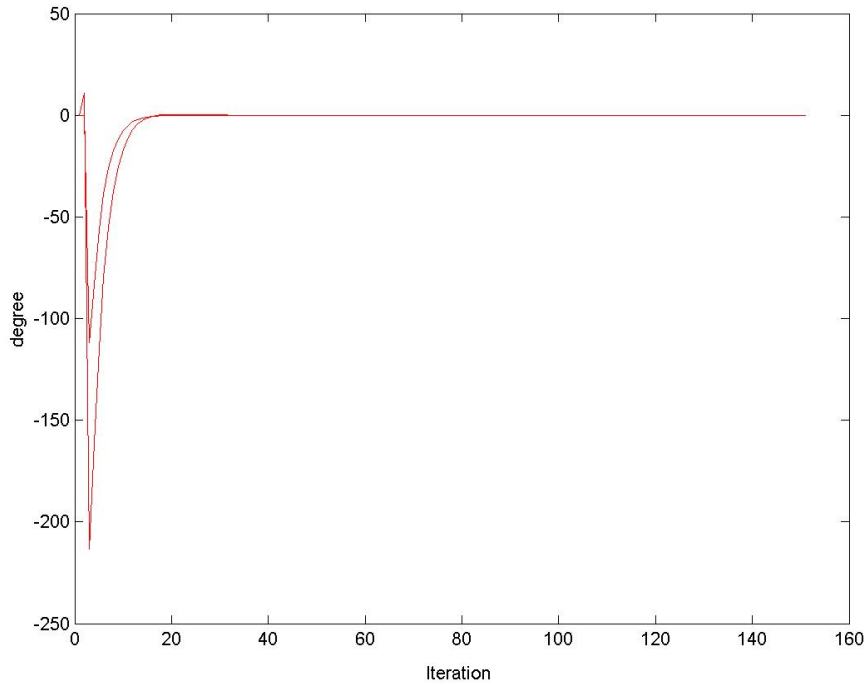

Figure 4.19: Results for the prediction of the heading. Plot of the errors in the heading values over time. See appendix A.23 for a bigger plot.

error	Means_{1-16}	Means_{17-59}	RMS_{1-16}	RMS_{17-59}
x	-51.3	0.09	83.1	0.21
y	-26.05	-0.05	42.51	0.14

error	Means_{1-16}	Means_{17-59}	RMS_{1-16}	RMS_{17-59}
x	-54.3	-17.87	56.53	27.34

Figure 4.20: RMS estimation of the positions and the heading itself in the heading prediction. After 60 iterations, the error converges to zero.

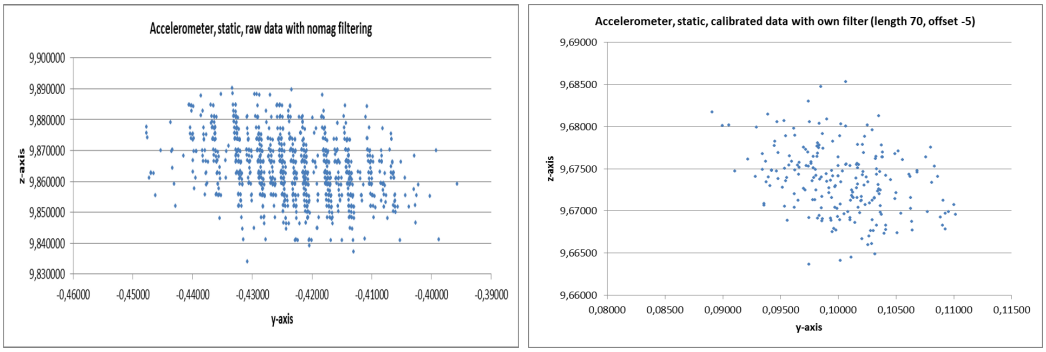

Figure 4.21: Sensor measurements in still mode. Left: Raw mode. Right: Noise reduction after calibration.

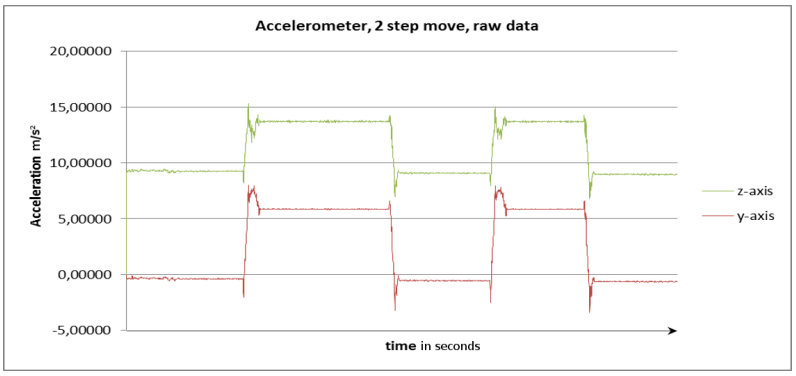

Figure 4.22: The following plots show a defined movement structure done by a PTU. This first figure illustrates the raw mode where the following figure show changes in the parameters of the noise reduction and the time offset.

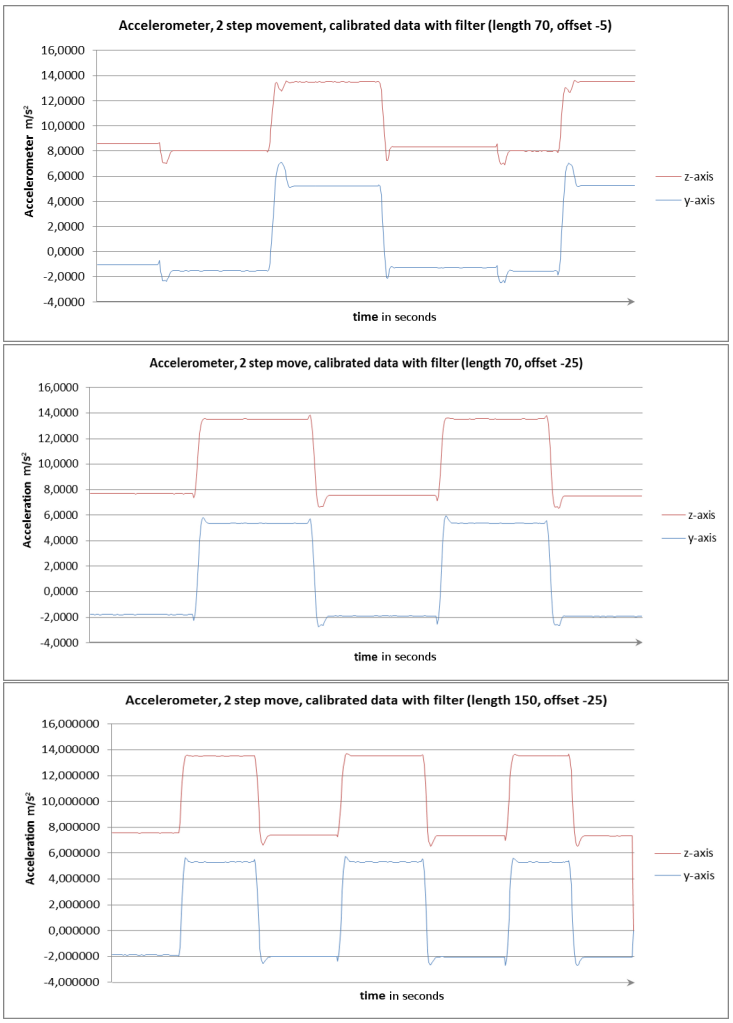

Figure 4.23: The plots show a defined movement structure done by a PTU. The figures show changes in the parameters of the noise reduction and the time offset.

RMS								all
raw	0.38	0.37	0.33	0.32	0.22			0.32
70,-5	0.11	0.61	.24	0.07	0.72			0.35
70,-25	0.01	0.17	0.2	0.39	0.22	0.24	0.31	0.22
150,-25	0.03	0.12	0.11	0.09	0.12	0.09	0.09	0.09

Figure 4.24: RMS for the individual intervals of the different filter parameter tests.

4.2.6.2.4 Heading Estimation

4.2.6.2.4.1 Heading By Image Processing The accuracy of the camera-based heading was analyzed using the PTU to make movements from -102.85 degree up to 102.85 degree around the vertical axis, see figure 4.25. The PTU was used with a horizontal step size of 5.14 degree. The resulting RMS of the noise of the marker detection was 0.01 degree, yielding an RMS of 0.03 degree for the rotation accuracy. The image in figure 4.27 shows a second test where the same object was repeatedly focused to measure the reliability.

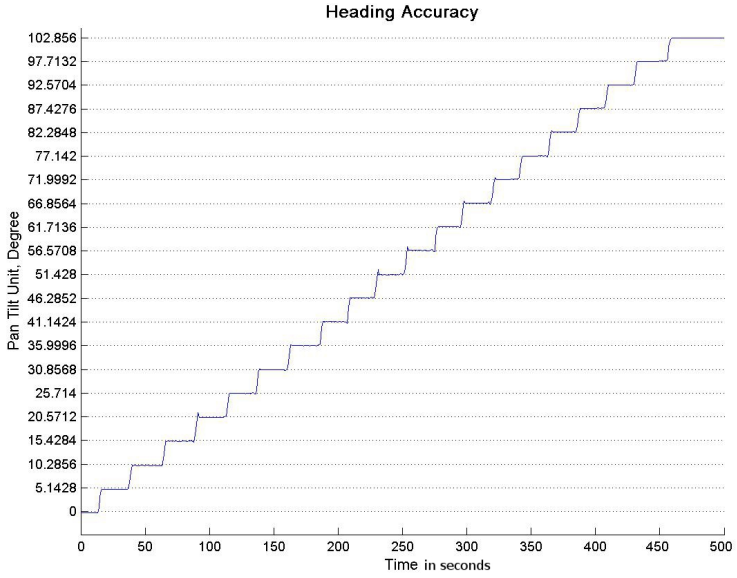

Figure 4.25: Measurement results taken by a PTU as Ground Truth.

error	steps	mean
with settling time	0.04 0.65 0.54 0.79 0.82 0.38 0.71 0.92 0.43 0.67 0.32 0.46	0.56
after settling time	0.01 0.03 0.02 0.06 0.08 0.05 0.04 0.06 0.06 0.03 0.08 0.03	0.05

Figure 4.26: Error estimation of the PTU test.

To estimate the performance of the system, the bearing of three real objects was taken for 10 times from a non-moving platform to analyze the repeatability, the noise and the resulting overall system accuracy. Figure 4.28 shows the true heading estimated from the positions in the WCS. Furthermore, it shows the means of the

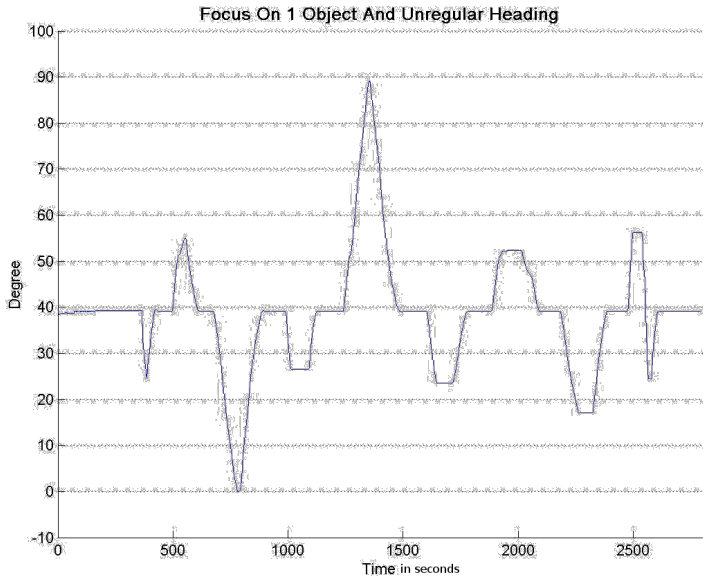

Figure 4.27: Testing the reliability by repeated focusing of the same object positioned at 39.06 degree.

10 measurements together with the standard deviations to the true headings. With standard deviations between 0.021 and 0.014 degree, the results are quite sufficient for nautical applications and rely to an error of 4 pixels in the augmentation which allows an overlay with 3D symbols.

Objects	True Heading	Means of 10 tests	Std. Dev.
1st	39.06	39.09	0.021
2nd	51.24	51.21	0.019
3rd	27.20	27.22	0.014

Figure 4.28: Measurement results for 3 objects, repeated 10 times; all values in degree

4.2.6.2.4.2 Heading By Inertial Sensors The same test has been done for the heading estimation by only the inertial sensors. The gyroscope were used to estimate the users orientation by using the angular rate of the turns. The test was done with several objects at the coast. There have been 5 objects with a change in the heading up to 130 degrees. The return path increases the time dependent bias in addition. The figures in 4.29 show the results. On the left, the mean RMS are shown. Within the series, there are big differences due to the drift which increases the error dependent on the rate of changes. After 5 objects, the device returns the

path which reduces the errors. The ending value should be the starting one, the figure shows differences of 21.89 degrees for a series which took 121 seconds. The errors are very large due to the long sequences and the high angular changes. In the field of the first few objects, the errors lie at 3 degrees. The figure 4.30 shows the results. If there is an recalibration in short time steps, for example at each object, if they are often observable, the errors can be minimized.

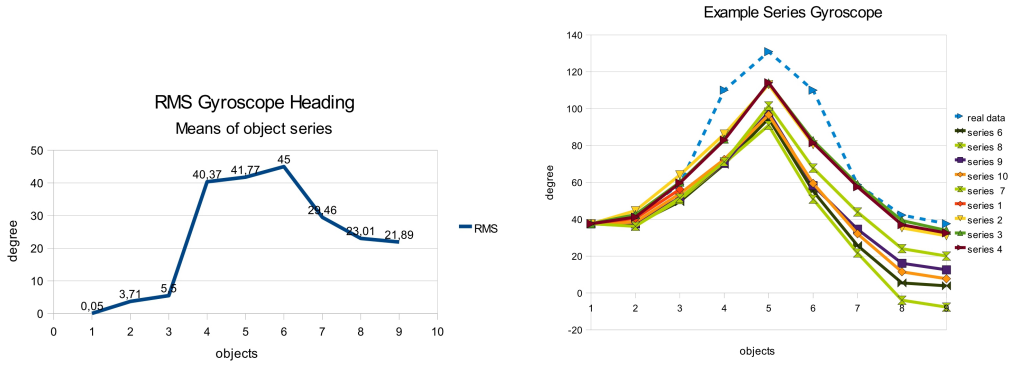

Figure 4.29: Gyroscope test; Many series of objects have been observed and the RMS calculated.

Objects	True Heading	Mean Heading	RMS
1st	37.58	37.55	0.02
2nd	42.26	39.44	2.73
3rd	58.79	55.4	3.2

Figure 4.30: Gyroscope test; RMS estimation of the first objects with lower changes (in degrees).

4.2.6.2.4.3 Range of Motion To estimate the movement abilities of the user, the plots in figure 4.31 show the test results for the translated device referring to the marker line. As the results show, there are absolute translation errors around 1.0 degree, but not dependent on the users rotations, only on the translation. This shows the restricted area for error free movements for the marker-based approach. To overcome this, see the improvements of the system, described in the next chapters.

4.2.7 Summary

The evaluations show that the best combination of sensors with the highest accuracy is the arrangement of inertial sensors and image processing. The low precision of the gyroscopes results of the drifting properties of the sensors. The marker-based approach is reliable and precise. The drawbacks are errors if the user move too

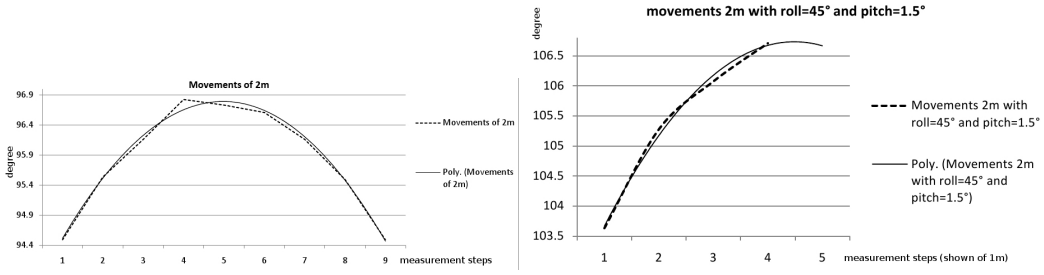

Figure 4.31: Measuring results for tests with and without rotation of the device while translating up to 1m in opposite directions in the area of the marker.

far away of the marker and the indoor characteristic of the application. The next sections will focus these topics and the missing interactive visualization and show howto overcome the drawbacks.

4.3 Charts Analysis

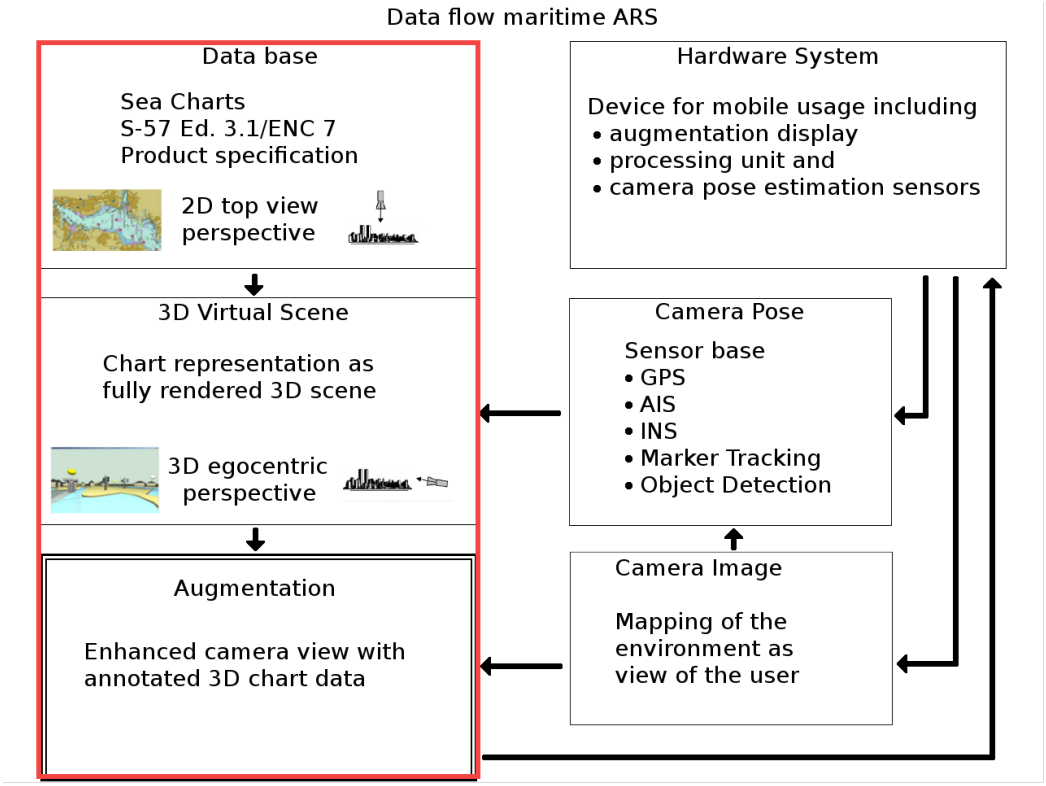

Figure 4.32: Overview of a maritime Augmented Reality System

In the overview in figure 4.32 the topic of the following sections can be found on the left side. To overcome the drawbacks mentioned in section 4.3.1 of the display concept of the sea charts, this section deals with a topology-building approach within the sea chart data. At first, an in-depth analysis of the data base is given. The analysis shows the deficits and possibilities for the interactive usage of a 3D sea chart scene.
The drawbacks of the fully virtual scene are many occlusions in the users view. This section analyzes the charts data concept to find aspects for the restructuring of the data scene referring to the requirements of an augmented view. After the explanations of the data concept, the developed approach to build an adequate data scene is shown.
With the help of this new structure, firstly, the user will be able to interactively change the scene to obtain information for only the interesting areas, and secondly, the object detection process will be able to use the object data meaningfully.

4.3.0.1 Visualization

The original visualization of the 3D module is a fully virtual scene. The figures 4.33 and 4.34 show different virtual scenes as an example. Unfortunately, fully virtual

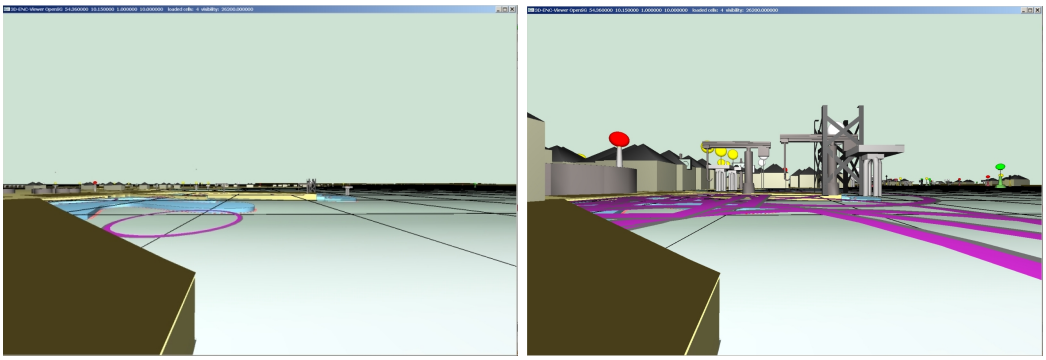

Figure 4.33: Visualizations of the original virtual scene.

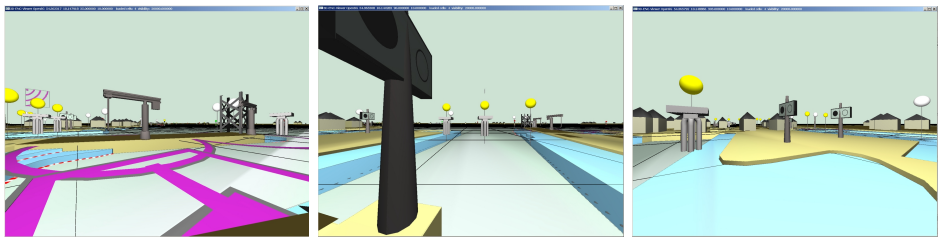

Figure 4.34: The original virtual scene. Three different views of the virtual camera. The user sees a fully rendered scene with covered parts like the sea surface or the natural background behind the objects. Objects can occlude each other significantly.

scenes mean completely occluded images in the view of the user. For the augmentation case, the scene has to be structured. In figure 4.35, on the right a screen shot from the new view through the augmentation camera with superimposed data is shown. The system can be used to mark certain objects and retrieve additional information from the sea chart database by pointing towards them. The similarity to a standard analogue binocular secures an intuitive usage. It needs no interaction devices except of the device motion. The figure 4.35 illustrates the possible visualizations. On the left hand, the image shows a fully rendered 3D scene from the 3D module given by the ECDIS system. The scene is not prepared for an augmentation which leads to a mainly occluded background scene. To manage this, the user can decide to choose display categories. This is shown in the middle of the figure. The image on the right is a screen shot of the users sight. The user gets information about single objects by pointing at them. All information given by the ECDIS system can be displayed in this way. Normally the objects are marked in the scene, but if it fails or if an object is missing, the user can also take the bearing of a real object and gets a choice of likely navigational aids.

Figure 4.35: Visualizations of the new augmented scene in different types. Left: The fully rendered scene is overlayed onto a video image. The water surface or other objects are not detectable anymore. Furthermore, the objects occlude themselves. Center: An adapted augmentation image is shown. The main orientation aspects like sky, background and ground are visible and furthermore, the interesting data is overlayed and conspicuous. Right: The user's view is shown. The augmentation scene is enhanced with blue buttons to interact with the scene and navigation information in the bottom like a compass, the original 2D sea chart or objects which follow when the user moves in the right or left direction.

4.3.1 Problem Specification for 3D Charts

The underlying ECDIS software provides a new 3D module. Laying upon the 2D data structure, the new module models all 2D chart data symbols as 3D objects for a fully virtual scene. This includes areas as water surfaces or water bodies. An example is given in figure 4.34. Here, the main point is, that the data structure is the same as in the 2D case and that the modeling is not optimized for an augmentation scene which is characterized by a simultaneous display of video signals. This holds for many 3D scenarios where the data is generated from 2D charts. The novel module of the ENC software is able to offer a 3D symbolization and a scene graph structure of the data. 2D ENC data is designed for a top view onto the map. A change of

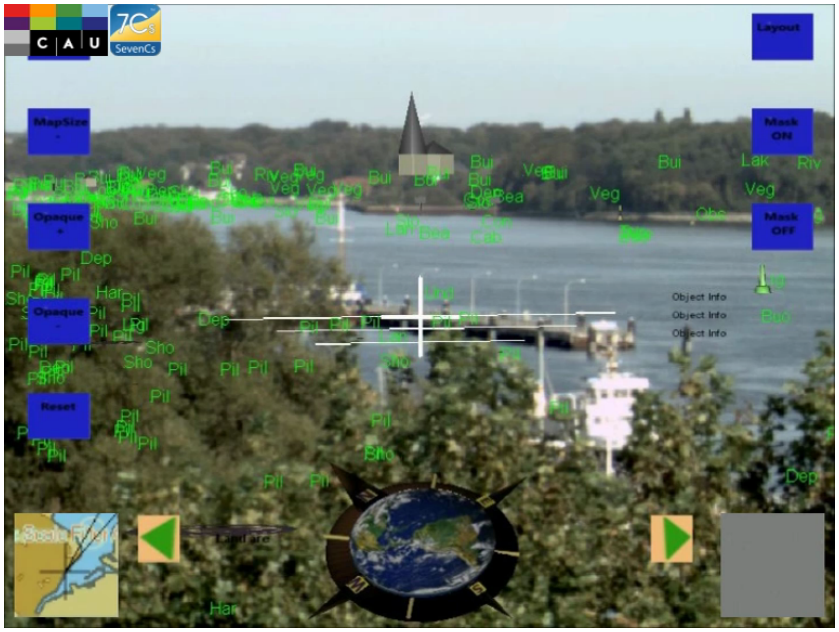

Figure 4.36: Even the overlay of text information leads to crowed displays.

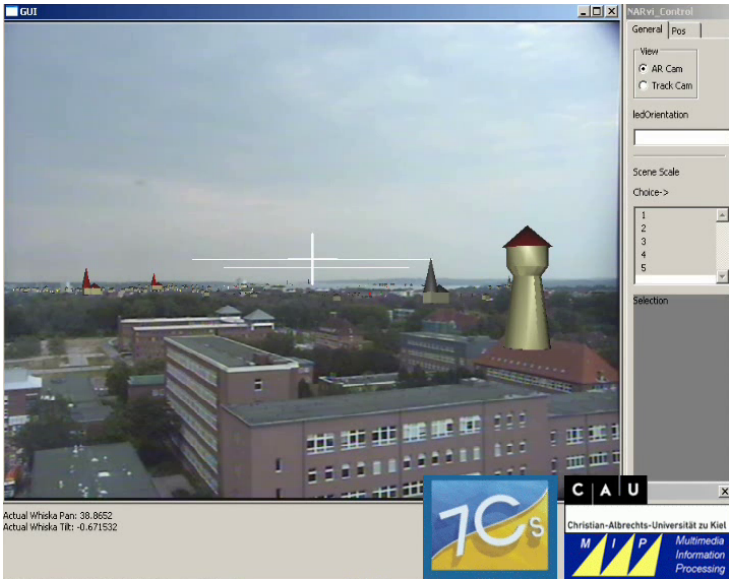

Figure 4.37: Small objects have to be overlayed with too big symbols.

the perspective into an egocentric one, where the user is mainly in the height of the objects, leads to significant occlusions. The data is concentrated at the coastline and the scene is rendered with a radius of 10 km. One example with almost only text information is given in 4.36. On the other hand, the symbols have to be bigger than in reality to be visible in the display, the example in 4.37 shows the context. Even displaying the safety information, which is defined as the minimum displayed data, may result in an overloaded view. It is therefore necessary to reduce the data in a way that the user can operate in all situations properly. In an interactive view, the user should decide which information is important at a time, e.g. the entrance of a harbor, but not the landmarks of the city in the background. To decide which properties of objects are used to filter the scene, the individual use case of the user should be known, which is not always possible. The aim here is to give the user semantic and local information to support a user-guided selection. In the 3D scene, the user can then select semantic groups and decide in this way which sub-levels are shown and which not. This enables a context-driven scale scheme in interesting areas of the scene, where at the same time other areas stay in an overview status. The user should always be able to exactly know what he sees and be guided through the data base by intuitive information.

4.3.2 Structure Analysis

4.3.2.1 2D S-57 Chart Standard

Standardized 2D chart data is optimized for the usage on a display which is situated perpendicular to the user. That means the user always stays in a top view perspective.
The data can be displayed in scaling groups which have to be chosen before a chart is symbolized. This calls for the knowledge of the kind of provided objects in a scaling group. In the case of a 2D chart display the rendering process is fast and a switch between the scaling levels is possible. The change in the scaling means for a 3D scene that the complete set of data, often represented as a scene graph, has to be exchanged. This is not an adequate data handling for a real-time application. The second point is that instead of the top view perspective the user is situated within the objects and the rendering of the users viewing leads to occlusions and inseparable objects. Occlusions are semantically defined and their definition changes with the focus of interest. In the classical approach there is no clear interaction possible.

4.3.2.2 Object Structure ENC/PS

The objects are defined by a separated storage of context and territorial information. This separation of the data is done to model the human object detection. Figure 4.38 shows a schematic overview of the structures.

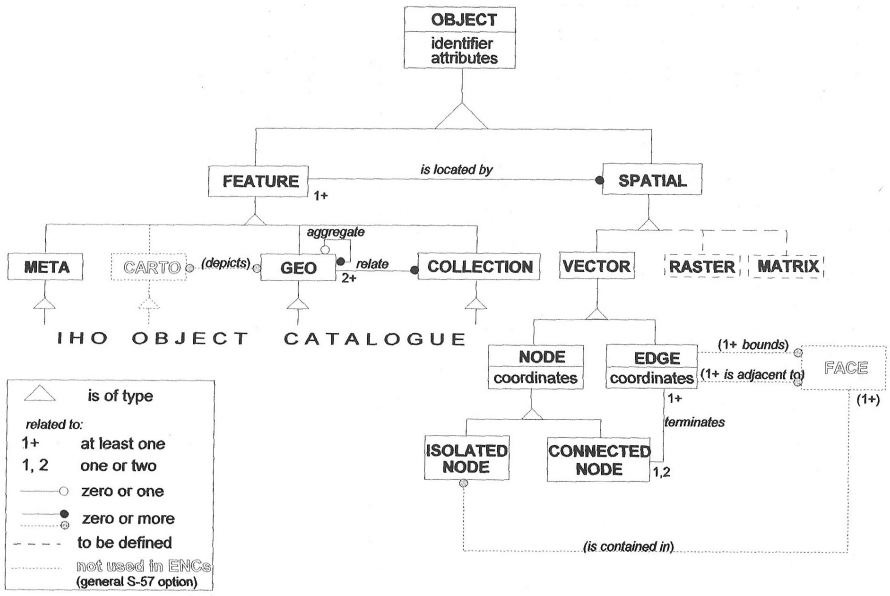

Figure 4.38: ENC data model. An ENC object is divided into the feature and the spatial objects. Additional relations are rarely set between objects. Relations can be built in the geometrical definitions between the point, line, area, node and edge structures.

The feature object holds the information of the functionality and specific contextual data. This includes the information about the standardized object classes. These classes are defined in the S-57 object catalogue.

The spatial object part stores the position and the display geometry. Until now, the geometrical modeling is defined for the 2D case. Possible representations are the vector, matrix and raster format. The ENC standard uses the vector-based modeling to describe the objects. In this case the spatial object is structured by a primitive and a segment type and uses line, point, area, point cluster, edge and node definitions. See Figure 4.39 for an example of the usage of the components within the chart data.

Due to the demand of the ENC/PS to reuse the geometrical features (chain node structure) a vector topology can be built, that can be provided by the ECDIS system. As different geometries share several geometrical features, the relations provided by this solution are for example area-to-area or edge-to-area relations. They can describe if areas intersect, lie within or if nodes belong to areas. Beside the single object definitions there are only two group definitions, G1 and G2. These groups of areas model the earth's surface and all others. All G2 objects are related to a G1 object, but within the G2 group there exits no structure.

4.3.3 Drawbacks

The scaling concept of the sea chart data implies that for a special region there exist several charts providing different data. Table 4.39 is a listing of several aspects of the same region of 5 nautical miles. The scene is represented by three scaling

	EC Harbour	EC Approach	EC Berthing
Cells	3	1	1
Total	8561	2622	532
P-Prim	3963	1259	401
A-Prim	2405	550	66
L-Prim	2077	630	61
S-Prim	116	183	4
Display Base	25*	10*	7*
SkinOfTheEarth	495	143	23
Our new Minimum	2597	783	37
named	539	280	46

Figure 4.39: Number of ENC objects (search radius 5 Nm) in different scaling levels (columns) differentiated by the kind of objects. P-Prim, A-Prim, L-Prim and S-Prim are point, area, line and sounding objects. Total means the sum of all 2D chart objects. The new minimum of the displayed objects in our 3D case is the sum of the display base, line objects and the minimum level in the new concept. (*dependent of the display filter)

groups showing a different number of objects. The scaling *Approach* fits here as an overview whereas the scaling *Berthing* is a detail chart of a harbor. Even in the *Approach* mode there are 2622 objects, mainly situated crowded across the coastal line. For each scaling level, the displayable objects change. This leads to an non-comprehensible placing of objects, as the user can not clearly observe the display structure. Even if the user would know, which specific objects are displayed in which level; if the user does not realize the scaling level he displays, expected and viewable objects may not be displayed. Furthermore, a quick change between the levels is not possible, because it leads to a complete new scene graph which has to be generated or all scaling level scenes have to be hold in memory. But to be able to support fast situation changes the user should be able to interactively choose the regional, interesting topics of a scene. In the sea charts there is no possibility to use view filters or filters which define these use cases. Supported by the sea chart system are only the scaling levels and the choice of single object classes. To handle the number of objects a hierarchy of the objects is built. This makes it possible to select the level of detail referred to the displayed information. Starting in an overview situation, the user can select the region of interest and guide through the display management.

4.3.4 New Charts Preparation Concept

To overcome the lack of a context-based display concept of the sea charts, this section shows how the data can be prepared to extract an object-to-object topology. This data preparation is not only useful in this context, other scenarios, e.g. in the GIS context advantages are the visualization of underground objects without the

occlusion of overground objects, the visualization of objects only laying on especially labeled areas like urban districts or the reduction of the so called "Skin of the Earth"-surfaces. This can be used in every geo-spatial 3D scene to restructure the scene for dimension reduction, hide mechanism during visualization and new semantic object categorization based on object geometries. See figure 4.40 for a schematic overview

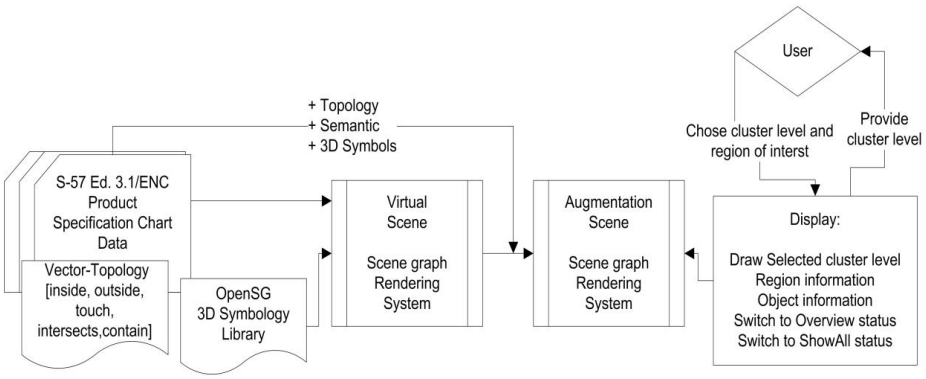

Figure 4.40: Data concept of the introduced display management.

of the concept. To generate the semantic information, the ENC data has to be preprocessed, as there are no hierarchical relations between the objects.
CityGML [1] is shown to be a good choice to store the generated information in an applicable format. In the next sections the conversion to this scheme and the data export into a scene- graph system, which is used to render the 3D scene onto the video signal, will be explained. This scaling concept exchanges the scale groupings of the IHO. The number of displayed objects can be reduced to 1/3 without loss of information. This is done by defining a hierarchy between the objects based on their spatial relations. Each subtree in the hierarchy can be displayed independently. By utilizing attributes of the objects, even semantic groups can be built. The following chapter explains the concept details.

4.3.4.1 Derivation of Hierarchy Relations

The goal of the approach is to define a hierarchy between the chart data objects. As the standardized data structure does not provide an object-to-object topology we need to define relations between objects.

4.3.4.1.1 Building Topology Within the geometrical definitions of an object exists an implicit vector-topology. We utilize the relations which provide an unambiguous hierarchy. Figure 4.41 describes the process. The drawback of this top down method is, that we can only process area objects. Due to the lack of information which point object refers to which area the second step is a bottom up approach starting at the point primitives. Each of them refers at minimum to one area, as all chart cells are assumed to be surveyed. Calling all areas on which a point primitive is situated leads to a number of unordered area primitives, except of the relations

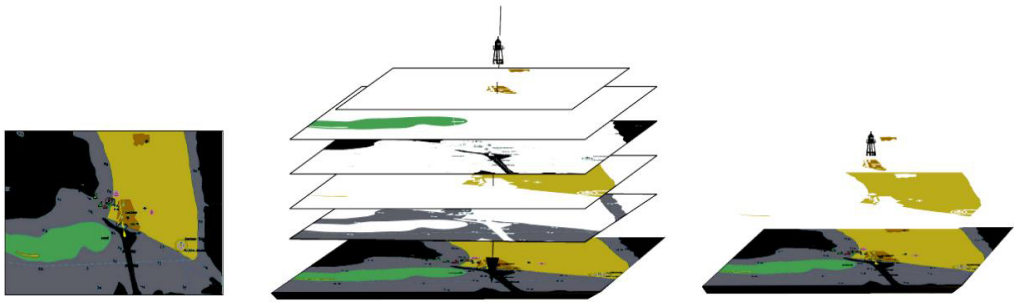

Figure 4.41: A general chart is interpreted as sectional model. The areas define layers by their territorial characteristics. Primitives are situated above the areas. Starting with the most important area the base hierarchy can be built up directly by the standardized chart data, evaluating the relations 'inside' and 'contains'.

to the G1 or G2 groups. To order the remaining areas we use the surface area, as we mainly want to relate objects due to their geographic context. This approach is similar to the base of the vector-topology, but in this way we can manage all objects of the chart data, including point primitives. After these estimations, each point primitive refers to a group of ordered areas. An example of the hierarchy is given in figure 4.42.

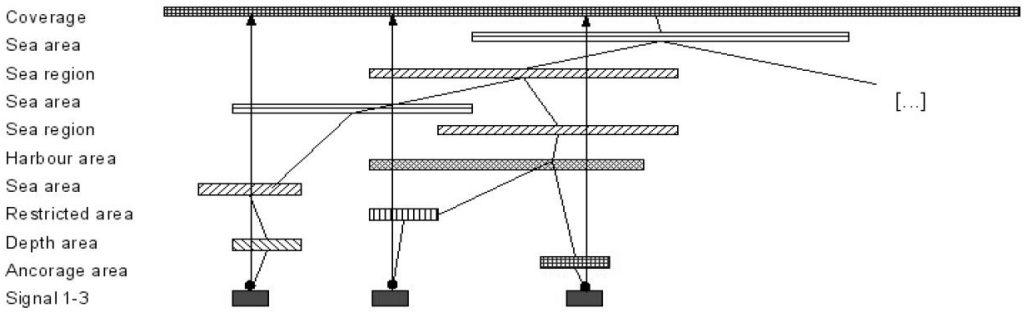

Figure 4.42: Example of the new hierarchy. The single real areas are mentioned on the left of the image. The new relations are drawn by the lines.

Excluded of this approach are the safety ("Display Base") information since they have to be displayed always and line objects which mainly do not refer to a single geographic region.

4.3.4.1.2 Building Semantics In figure 4.43 scene graphs generated by the virtual (old) and augmentation (new) scene are shown. The left graph shows, that the advantages of a scene graph system can not be used here. The data used is the original 2D chart data. This resolves in one scene node which holds all objects (here about 20.000) without any structure. The new object-to-object hierarchy is

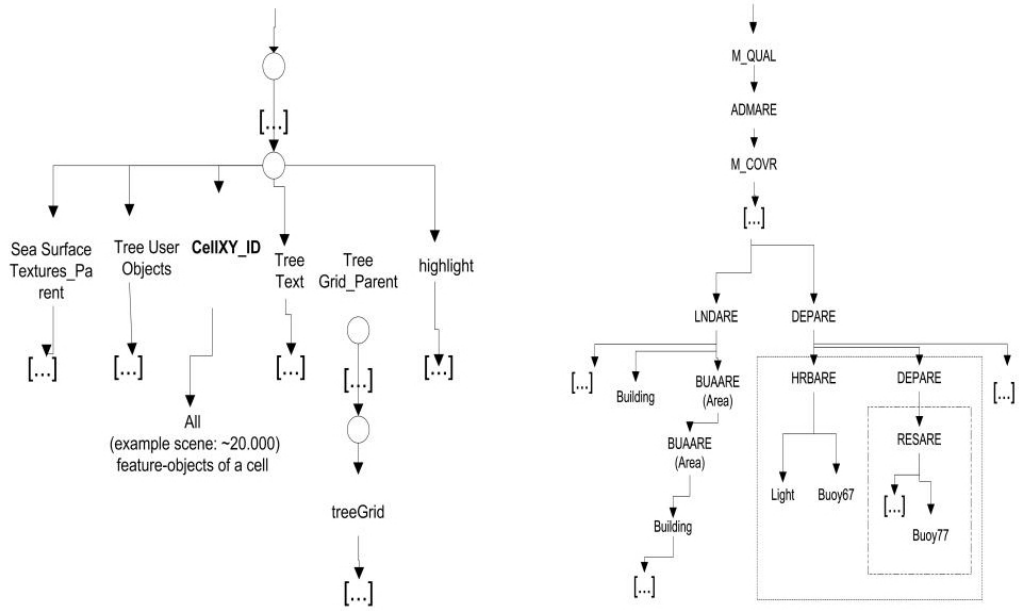

Figure 4.43: Left: Old data structure. Right: New data structure.

illustrated on the right side. The important areas are situated in the upper parts, the point objects like sea signs and signal stations can be found at the bottom. All sub trees can be activated or deactivated. The dotted lines represent semantic groups, which define the selectable sub trees for the user. As the data has a realistic instance we can interpret the objects within the hierarchy. To use this information in the display management, the generation of additional semantic groupings is done by utilizing the S-57 object attributes. Sub trees in the hierarchy referring to a labeled node are used to define a semantic group. As the figure 4.43 shows, semantic groups can have a topology itself. To show the number of possible semantical groupings, see figure 4.39 for the evaluation of the attributes ObjectName (OBJNAM) and NationalObjectName (NOBJNM).

4.3.4.2 Standard Conformity

The ECDIS software system models relations between objects, which are normally not filled. These relations are used to store the generated hierarchy into the 2D chart data. As only the value of the relation is changed, neither the size of the data nor the standardized structure is affected. The approach can be used as a preprocessing, done only once. The charts can then be used for all visualizations by evaluating the relations. This can not be done for the semantic information we built. The solution here is to use an exchange format to store the generated information permanently until its usage in a scene rendering system. For this purpose a format is needed which can hold not only the semantical grouping but also the standardized 2D chart data. Since the new Version 4.0 of the Standard S-57 will use a defined GML

(Geographic Markup Language) Scheme called ($S57/GML$) as exchange format the *International Standard CityGML* of the OGC (*Open Geospatial Consortium*) fits the requirements very well. CityGML is a representation scheme for 3D scenes and can hold the 2D data by using only a few extensions of the CityGML standard.

4.3.4.3 Approach to Map ENC Data into CityGML

To overcome the mass of data and information, we aim for a hierarchical data structuring, which allows the possibility of object groupings by semantic. An existing method of filtering objects in ENC data is to select object classes, e.g. show buoys, which has no semantic reference. A hierarchical structure of all primitives, e.g. buoys, by exploring the relation schemes implemented within the geometry definition of the objects have been built up . It is defined in the S-57 Standard that geometries should share their segments. This allows the ECDIS system to provide a knowledge base if primitives e.g. "touch" or "intersect" each other. At a higher level, there exists a second relation scheme which relates the complete objects to each other. Using these relations which are modeled but not utilized (e.g. "has-a"), leads to a storage of the hierarchy within the data file without increasing the storage size. That does not apply for additional information which is generated by the identification of semantic groupings. For this data we need an applicable storage format. Semantic information can be built by taking hierarchy levels and with the help of its children and by neighborhood relations between these groups, the topic of a cluster can be found. Topics are e.g. object names of land areas or sea regions. As a result new requests are possible like getting all objects of an entrance of a harbor or of a build up area. In combination with the given clusters, the filtering by the object classes is still possible. The approach here is not to map all ENC attributes in detail to the CityGML structure, but store the extension of the 3D ENC data format. These extensions are the hierarchy within the ENC objects and the semantical groupings. The creation of information leads to a need for another storage format as knowledge representation. CityGML provides the structure for thematic modeling, semantic descriptions, appearance storage and 3D geometries [109].
This offers the infrastructure for a 3D ENC scene. It fits very well to the object types given by the ENC object catalogue since the complete scheme of CityGML only differs in 10 sub-definitions. CityGML is an application scheme of GML3 (Geography Markup Language). It was developed since 2002 and follows the ISO 19100 like the S-57 standard in his next Version 4.0 (S-100) will also do [7]. Furthermore the new S-57 version will open the possibilities for 3D geometries, time-varying objects and introduces a new S-57/GML scheme for (2D) data exchange. The standard serves as a base for the product specification ENC, but will not be published before the end of 2012. Beside this, there exists an XML scheme for 2D ENC-data, called MarineXML [4]. This shows, that there is a development to 3D modeling and XML as exchange format, but a scene graph structure needs in addition a hierarchy. CityGML provides an extension scheme called *application domain extension* (ADE), with some ADE already successfully implemented and available [191]. Therefore an ADE for the use of CityGML with standardized ENC data was created, called ADE

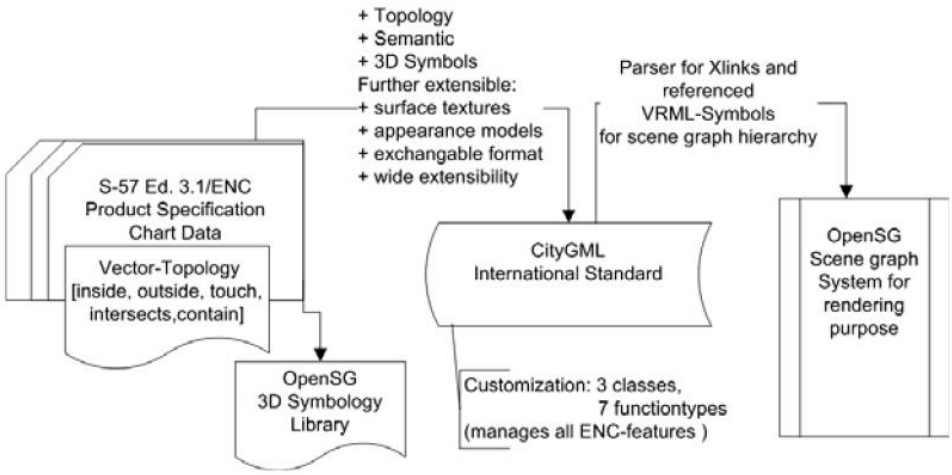

Figure 4.44: Data flow of the conversion from S-57 ENC data to CityGML and into a scene graph system.

Enc. Figure 4.44 shows an overview of the data flow.

4.3.4.3.1 CityGML ADE Enc Definition Figure 4.45 shows a few examples of ENC objects and their conversion into CityGML. CityGML has 11 thematic models, from which four had to be extended. Main traffic structures like traffic directions, tracks and routes are mapped to the "Transportation" module. Single features which have a local meaning and artificial sea signs are given to the "CityFurnitures". Water elements, vegetation and buildings are given to their corresponding themes.

Mapping ENC <> CityGML

<<extract>>

ENC object	corresponding CityGML object
BCNCAR	Module: CityFurniture, Class: Traffic, Type Roadsign
BOYLAT	Module: CityFurniture, Class: Traffic,Type Roadsign
CURENT	Module: WaterBody, Class: Stream
DEPARE	Module: LandUse, Class: Water, Function: Sea, Usage: Military
FERYRT	Module: Transportation, Class: Waterway, Function: Ferry,
FSHZNE	Module: WaterBody, Class: Sea, Function: Waterway, Usage: Fishing water
HRBFAC	Module: Transportation, Complexclass: Waterway, Complexfunction: Habour
LIGHTS	Module: CityFurniture, Class: Traffic, Type: Roadsign
T_NHMN	Module: WaterBody, Class: Tidal water, e.g. Waterleveltype: Mean high tide
LOCMAG	Module: LandUse, Class: Water, Function: Special Function Area
OBSTRN	Module: Cityfurniture, Class:traffic, Function: others
CTNARE	Module: Transportation, ComplexClass:Waterway, ComplexFunction: Traffic marker

Figure 4.45: List of few examples which shows the mapping of ENC data into CityGML objects. A full list can be downloaded with the ADE Enc.

The figure 4.46 shows the extensions of the schema for the objects which could not be mapped clearly. The ADE Enc can be viewed[2]. The extensions are additional definitions given by the "Generic Property"-attributes of the shown objects. The figure shows the kind of definition, the number and the mapped ENC object type. Additionally, all of the shown classes and the "Building" module with an implicit geometry tag were extended. Especially for "Building" it seems to be a good supplementary concept to use predefined models, because buildings often look similar and sights could be modeled once in detail. The semantic groupings are mapped by defining "CityObjectsGroups", which are designed to group objects.
Currently the 3D ENC module only supports one level of detail (LOD) which results in an unutilized LOD concept of CityGML in this case. Furthermore, the ENC feature-IDs were mapped as object names. The object attributes of the ENC features are given in raw mode into the body of the defined objects. Important attributes which match the CityGML structure like position and color are processed independently. To integrate the generated hierarchy into the GML structure, the CityGML "Xlinks" scheme is used. The root object refers as a beginning and children are associated by the one-directional "Xlinks". The geometry is set implicitly, as the VRML file format to represent the 3D symbols of the ENC data was used.

[2]ADE enc: http://www.mip.informatik.uni-kiel.de/tiki-download$_file.php?fileId = 1318$

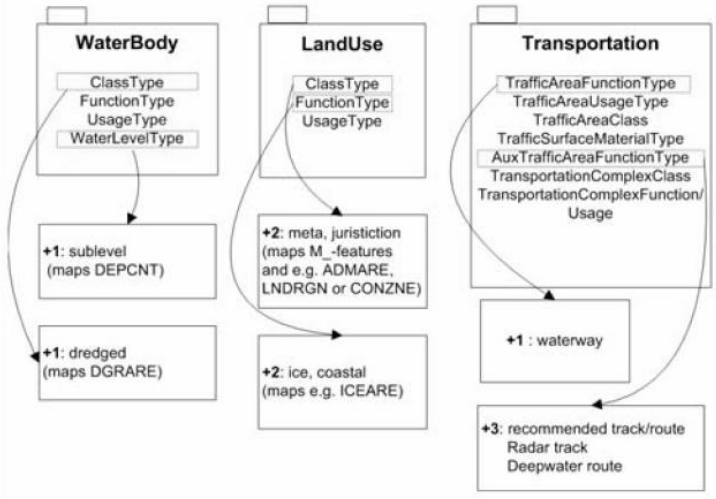

Figure 4.46: Extension principle for the ADE Enc. Only a few additional definitions have to be added.

4.3.4.3.2 Scene Generation New information from the ENC data have been generated and stored into the CityGML.

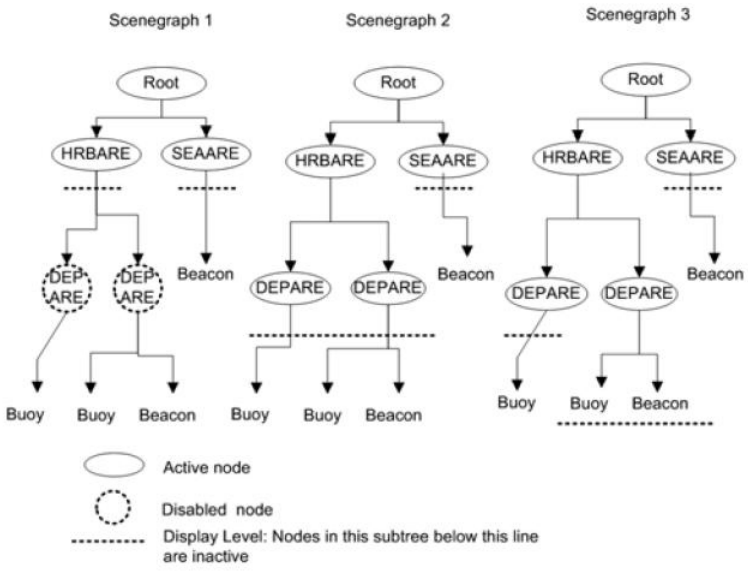

Figure 4.47: Principle of the display management. The scene graph is shown with different display levels, that decide which nodes are given to the renderer. To render the scene, the structure of the data and the data itself has to be exported to a scene graph.

Given a CityGML structure, which is enhanced with the described "Xlink" hierarchy, a parser just begins at the root node of the CityGML objects and detects the

transformation and the referenced VRML file. By relative transformations to the children, the tree-like scene graph structure can be built. Figure 4.47 shows three scene graphs with different display modes to show the display management. The semantic groups are modeled as special nodes in the tree. If the user chooses a semantic group to be interesting, the according sub tree is activated and can be shown, as requested. Each semantic node in the subtree can then be used to set the new interest limit. These limits are used to decide, which nodes with the appending subtree are drawn. Safety information and objects like tracks are not given to this display management, as they are not locally bounded or should always be displayed. Figure 4.48 shows the raw user view of the device without any overlays and the overloaded situation by drawing all information at a time. In contrast to that, the figure 4.49 shows the processed scene, where other LODs are shown. The left image shows that it is possible to decide e.g. if the sea or the land areas are important. The image on the right shows the next more detailed LOD for a region on the land area. This mode is able to display only the information which is selected. In the application mode the scene starts in an overview mode and the user can pick regions of interest and increase the LOD while unselected regions stay in the overview status.
A drawback of the 3D ENC module is the simple structure of the symbols, as there are only few surface textures. To improve this, Computer Vision techniques could supply more information like colors and textures [58], [209]. Furthermore, implicit geometry handling is used. If textures are changed, the referenced geometry file has to be updated. A next step could be to rebuild the referenced models in CityGML by its GML3 geometry structure to have full access to all properties.

4.3.4.4 Exploitation of New Structures

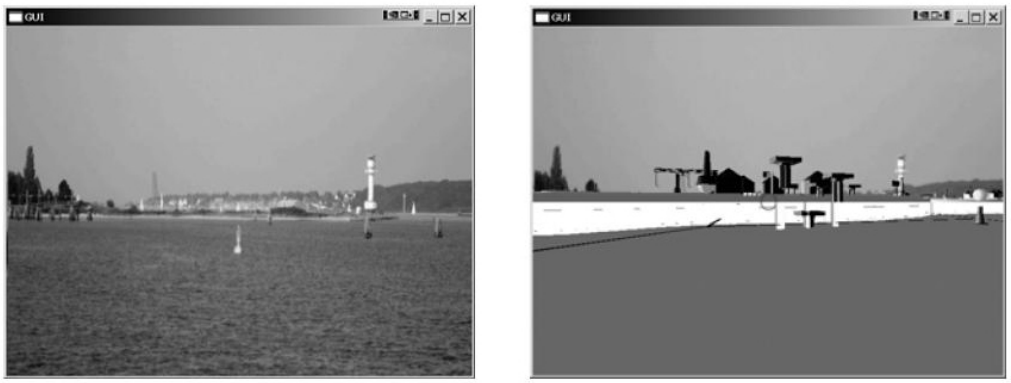

Figure 4.48: Augmentation scene with different LODs. Left: Nothing overlayed. Right: Everything displayed.

The 3D visualization is now leveled by the given hierarchy. This hierarchy can be handed to a scene graph rendering system. The user can decide if land or water areas are relevant. The changes are handled in real-time. Reduced data is marked

and the underlying objects can be displayed by selecting the group. Figure 4.51 shows 2 example scenes. In the top row sea signs (a building and a tower) occlude interesting signs on the water area in the background. The top right image shows the reduced scene without the occluding objects. The areas in the near of the user are suppressed. From the second row on another scene is shown where the user selects the focus. Four images are where the user selects all areas, only the covering areas without objects, only the land area or only the area of the marina. In the figure 4.50, the images are in a classical top view as in the 2D case, but only to show the behavior of the areas. The overview shows the reducing to interesting areas with their objects in contrast to the classical usage with defined LODs and only object class choices without geographical context. The results show that new object classifications are possible since they are now stored in labeled groups.

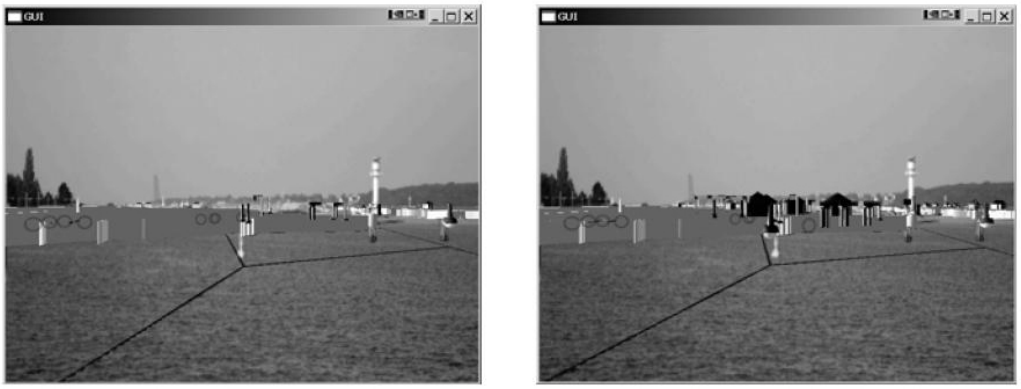

Figure 4.49: Augmentation scene with different LODs. Left: Land area chosen to be displayed while the water areas are not important. Right: Same land area with more details.

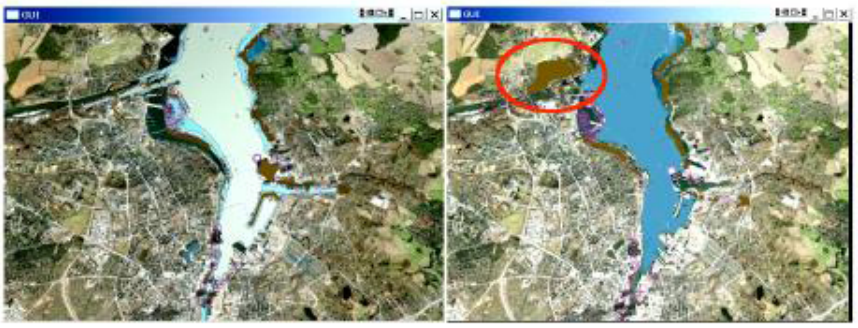

Figure 4.50: Left: The scene with only water areas shown. Right: One selected build up area is shown, where only objects to which the area refers to and display base objects are rendered.

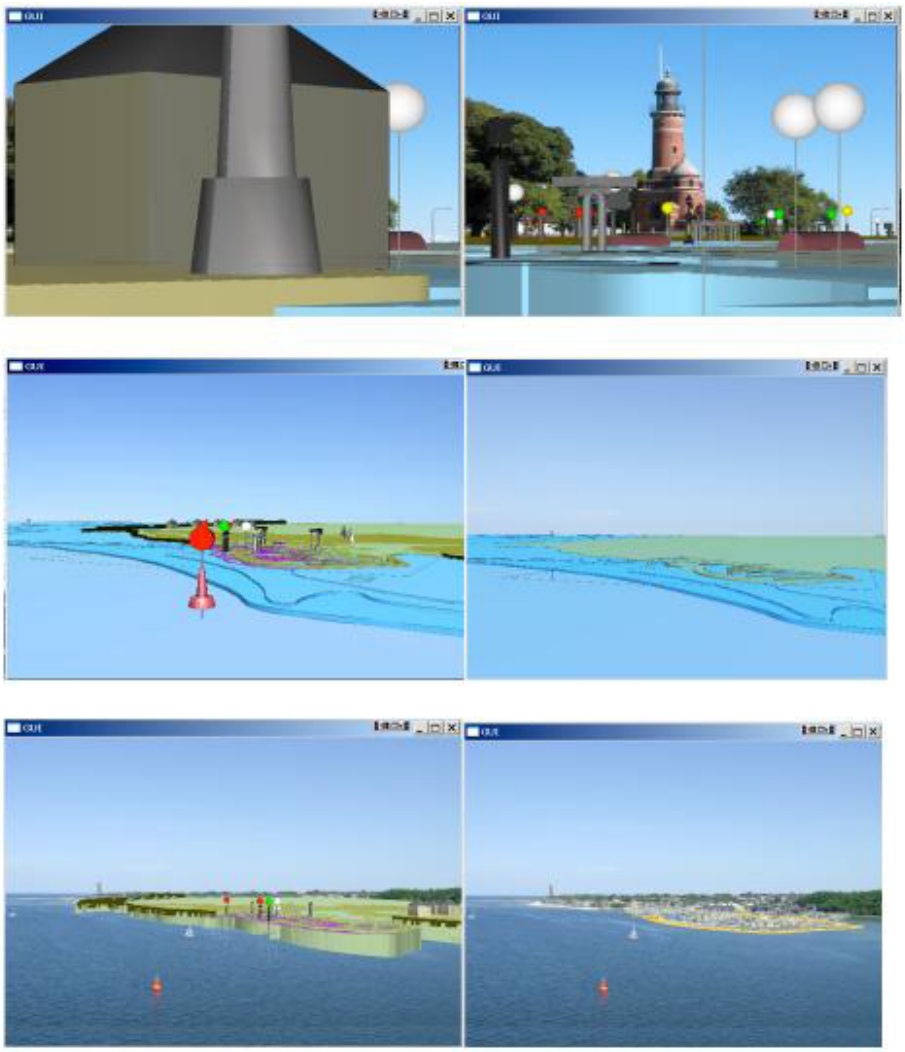

Figure 4.51: Top row: Occluded scene on the left and the same scene reduced on the right. Remaining images: Fully rendered, only 'Skin of the earth'-surfaces, single land area and the shoreline construction of a marina.

4.4 Probabilistic Modeling of Chart Knowledge

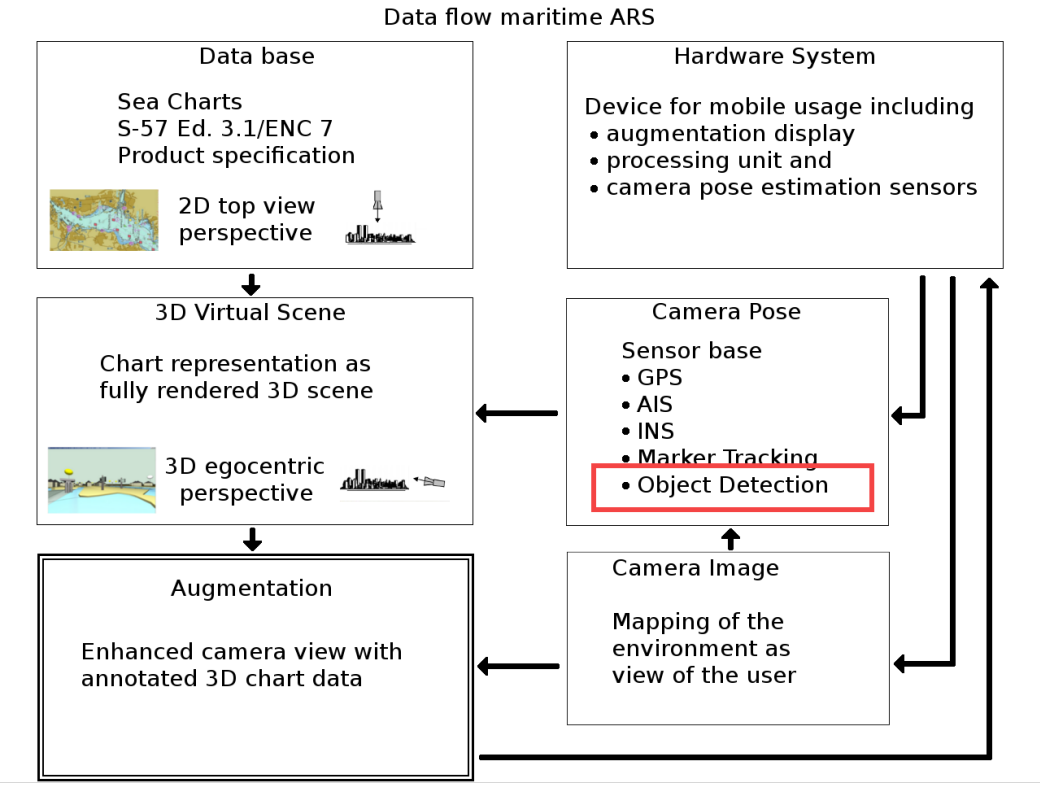

Figure 4.52: Overview of a maritime Augmented Reality System

To overcome the drawbacks of a marker-based approach by the benefit of sea charts, this section deals with a markerless solution to allow the outdoor usage of the AR device. The chart preparation mentioned in the last section is very crucial, as it not only allows the interactive usage of the 3D scene, it also simplifies the classification of objects by providing context-based information. The overview in 4.52 shows the aspects of the maritime ARS. The main developments are explained, but the basic device refers to a marker detection approach which reduced the movement range of the user and the usage to indoor locations. The question was, if there is a possibility to further improve the concept.
The aim in this section is to define a Computer Vision technique that allows the system to detect single objects in the environment to find the correct pose. For an image analysis approach a priori appearance-based information is needed to guide the detection process.
The device should be able to be used in unprepared environments which means in case of a ship, that the device will be used everywhere on the bridge even on the outside. Due to the mass of data of different coastal and sea areas, the training time and database space has to be minimized. It is not manageable to take training samples of each federal restricted area. Solitary in Germany, there exist

several federal administrative districts at the coasts which can act autonomously in defined purposes. Due to this, it is recommended to use more general information, e.g. provided by sea charts. By the help of a priori information the image analysis approach can be guided to detect interesting objects. The approach has to be able to estimate markerless the pose of the device, even for small objects far away in an every day situation. By using the general chart information the approach can only classify object classes, that do not suffice for a pose recalibration. There has to be a second approach which votes for single objects due to the classified classes. With the inertial sensors of the device, the viewing direction can be roughly predicted. To refine this, the charts must be able to provide appearance-based information of the objects by it's geographical context. This leads to the possibility to identify single real objects by probabilistic methods to recalibrate the pose.

4.4.1 Usable Object Information

Beside the knowledge of semantical relations between objects, information about the optical appearance of navigational aids is very important to define a search pattern for the image analysis technique. The ECDIS sea charts provide some information about objects which can be partly used in this context.

4.4.1.1 Navigational Aids

The S-57 conform sea chart catalogue provides 159 object classes with 187 possible attribute classes [31]. The object classes can be divided into the topics topography, hydrography, navigational aids and boating sports. The figure 4.53 shows the relationships in an overview. Objects related to topography are natural circumstances that can be used to navigate or dedicated buildings that can be used as a landmark but are not placed by nautical staff. The hydrographical topic belongs to sea-related objects like depths, soundings and wrecks. The navigational aids stand for artificial aids which have been placed to mark special areas like beacons. Attributes range from information about name and object type, position and position accuracy, the dimensions of an object over color and color patterns to survey dates. Which attribute is optional and which is determined for an object class is defined in the standard.

4.4.1.2 Useful A Priori Appearance Information

There are 52 classes of navigational aids, for which appearance information is provided. Appearance-based information in the ECDIS system are given by the classes color, color pattern, shape and height. All interesting classes with their appearance information are listed in the table A.3 which can be found in the appendix A.1.1.
Navigational aids can be grouped into two classes: artificial and natural/reused aids. Artificial signs are human made and situated at important positions for the navigational purpose. These signs are especially designed as navigational aid. The navigational aids of the second class are natural features like a group of trees or a river estuary or they are visual conspicuous like a windmill. These features are not designed for this task and they are rarely colored and mostly not in standardized

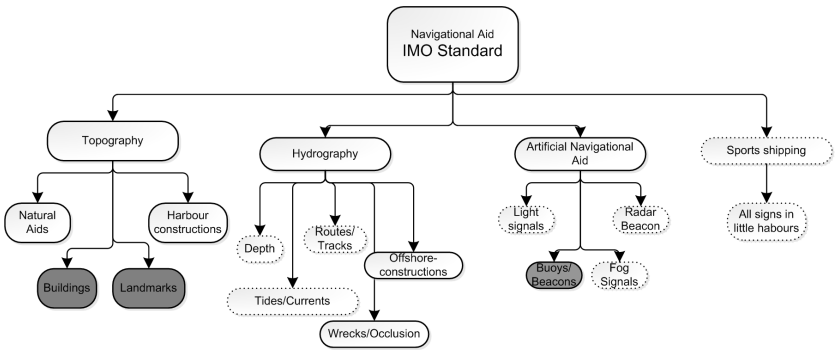

Figure 4.53: Overview of the catalogue of sea chart objects. Colored objects refer to artificial objects for which appearance-based information is given. The second group is marked by a fully black line. These objects are observable, but no appearance-based information is given.

Figure 4.54: Images of variations of navigational aids in different geographical zones, images by Heinz-Dieter Kuhnke.

colors. The basic information about these two classes varies largely. For the artificial aids there are many attributes standardized, but the reused features vary in each instance of an object class, e.g. bridges and there are no standardized class information given in the charts. We define artificial nautical aids as especially built and positioned navigational aids like light vessels, light floats, all kinds of beacons and all kinds of buoys.

4.4.1.2.1 Artificial Aids For the ECDIS system there are 13 color classes given: white, black, red, green, blue, yellow, grey, brown, amber, violet, orange, magenta and pink. The usage is defined for 52 object classes. The main colors of artificial navigational aids are red, green and yellow. Additionally to the mentioned base colors there can be used a second attribute called color patterns. Defined are 6 classes: horizontal stripes, vertical stripes, diagonal stripes, squared, stripes, border stripes and they can be used with 37 object classes. These definitions can help to classify buoys and beacons, but they do not take impurities, abrasions and regional changes into account.

There are 8 shape classes given: conical, can, spherical, pillar, spar, barrel, super-

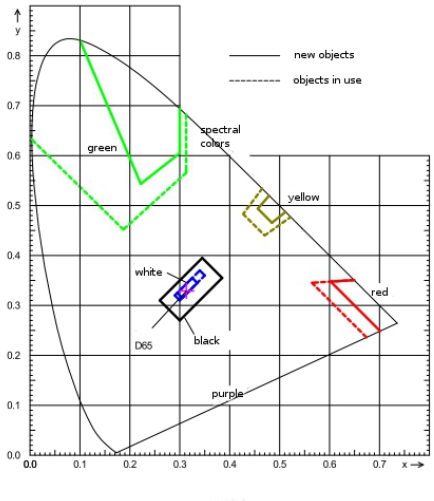

Figure 4.55: Color chart in the XY space for new (*straightline*) and used (*dottedline*) navigational aids. The colors green, yellow, red, white and black are shown in both conditions. The reference white point is D65. For detailed values see appendix A.1.1

buoy and ice buoy. These shapes are recommended. This attribute can be given to 7 object classes, which all refer to buoy types. Figures 4.56 and 4.57 show the possible shapes and real world examples.

An important group of aids is the buoyage system, defined by the *International Association of Lighthouse Authorities*(IALA). The standard is shown in the appendix A.4. The standard defines colors, shapes and color pattern for the different purposes of buoys. For some kinds of purposes the kind of buoy is free of choice. For the colors of buoys there exists a color specification from the *Federal Waterways and Shipping*

Figure 4.56: Comparison of symbols and real instances, images by Stefan Görrissen

Administration of Germany[3], which is recommended as technical standard. That means these colors are standardized by possible variations. The definition of the colorimetric requirements is given for swimming aids, poles, top shapes and visual marked radar reflectors. The figure 4.55[4] shows the color definitions in the XY color space. A table of important attributes of color and shape and their distribution can be seen in figure 4.58. The probabilities of the attributes are well-balanced and promise to be a good feature to work with. As a result, there are 13 colors, 6 color pattern types and 8 shapes without top mark shapes for 52 object classes. Due to the very small mapping of the objects in the image, shape analysis and very detailed pattern analysis approaches will not work. The aim will be to detect the colors in the image at likely positions. With the help of the position of the user in the chart and the relations between the possible objects, the most likely object classes should be voted. The local distributions of the object classes will help to achieve probabilistic a priori information for the voting of single object instances. By defining the

Figure 4.57: Symbols in real images. This quality will be the input for the image processing pipeline due to the zooming lens of the camera and the low resolution.

kind of estimators for the colors, the color pattern and the shape, the probabilities could be mapped to a Bayesian network to infer the best classification for a given segmentation or other statistical approaches.

[3]FVT Koblenz
[4]issued by WSV, Federal Waterways and Shipping Administration, 2010

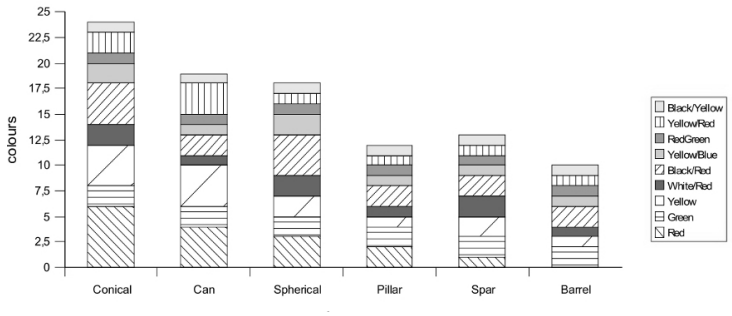

Figure 4.58: Distribution of shapes and colors for an example scene.

4.4.1.2.2 Dedicated and Natural Aids There are no special appearance-based information for objects of this grouping. They can be vegetation, a river estuary or buildings of unknown shape. Due to this fact, it would not be possible to detect these objects with a priori information. The strategy should be to learn the appearance of such objects. As the device needs a pose estimation before a position of an object can be clearly voted, the solution is to firstly detect artificial navigational aids which are defined in their shape and color and then to predict the position of the objects to learn. An analysis of the predicted image areas can define individual patterns to learn. Possible approaches to this topic are feature-based learning approaches or segmented adjacency graphs.

4.4.2 Concept

This section deals with the object detection for an ARS in maritime conditions. The detection approach uses probabilistic methods to benefit from the prior information given by the underlying sea charts. See figure 4.59 for a detailed data flow overview. At first the chart data base is used to receive color, color pattern, shape and hierarchy information about the artificial navigational aids. These information are used to define the structure and basic parameters like color models within a BN. At a second step, the image given by the ARS with roughly estimated pose, is given to a detector pipeline to describe the content by color and color patterns.
The single results of the detectors are given to the BN to iteratively infer the class of the observed object in the image. If the first detector leads to a very high probability, the detector pipeline can be discontinued to save processing time, if the sum of detectors does not reach a satisfying probability, the method has to be repeated. After the classification into classes, the information is used to build dynamically a second BN where the structure is given by the local environment information about the class and the neighbors of single class objects. If more than one object has to be surveyed, the second BN will infer the single class results and use the information of the user movement to model the probability of object occurrences.
After the identification of a single artificial navigational aid, the ARS gets a pose update and the system transfers the state into a probabilistic tracking approach.

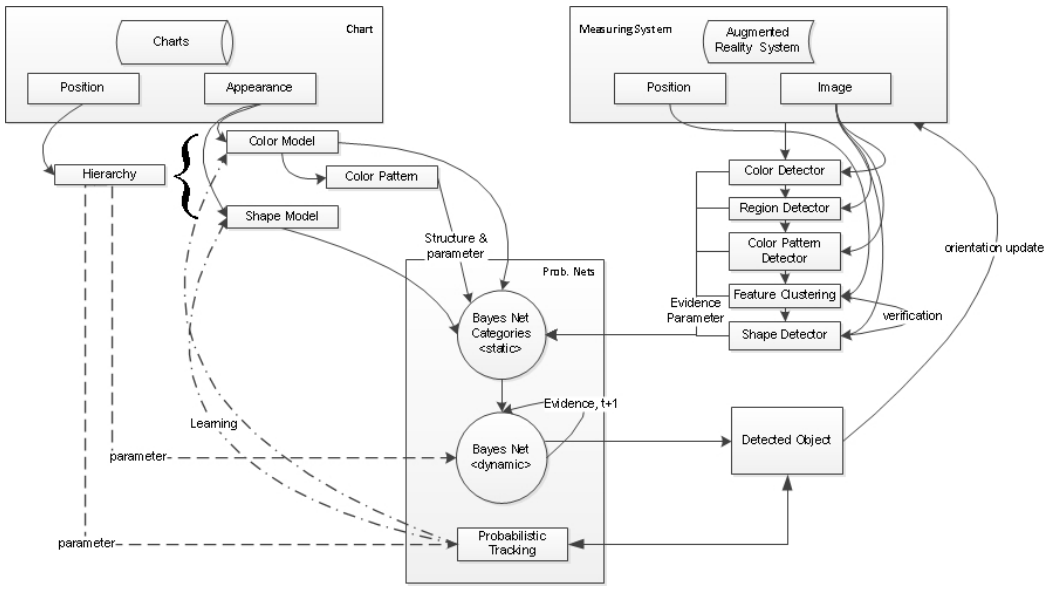

Figure 4.59: Overview of the probabilistic concept of object identification with prior information given by sea charts.

With this method, the occurrence of objects can be predicted which significantly improves the performance of the real-time device and allows more costly scene interactions of the user.

The resulting BN structure is illustrated in figure 4.60. All dependencies, a priori information and evidences are shown. On the left side the prior information, modeled by the distributions $\lambda_{1,2,3}$, is the result of the analysis of the sea charts. The right side is used to provide the evidence information a, b, d, e given by the ARS and the image detectors. The boxes P_i, R_i, R_j, O_i model the sub models of the structure, where P_i covers the pixel wise process, R_i the process per regions, R_j the process by clustered regions and O_i the process per assumed object. The clustering of regions to an assumed object is guided by the classification of main color o_i and pattern color $o_i^{'}$. The additionally observed pattern sequence leads than to an unified voting by $o_{,}^{'}$. The first summarizing box describes the first level BN, which models the chart knowledge by a static structure. The outcome is the class of the observed object o_i, if the probability reaches a high value. The second level BN is used to guide the detection process over more than one time step.

A second usage of this BN is the light-weight tracking after the positive detection of an object. This second level BN is built up dynamically during the processing time as it depends on the voted classes and the actual position in the chart. The input values are the best voting values of the observed object classes at the time t. The possible object instances and their geographical neighbors N are voted and the voting will be continued at the time step $t+1$ until an object is classified with high probability. In the case of the light-weight tracking approach, the BN is used to predict the next objects in the images and to improve the performance of the object identification process. The dynamic BN used in this concept predicts the next 5

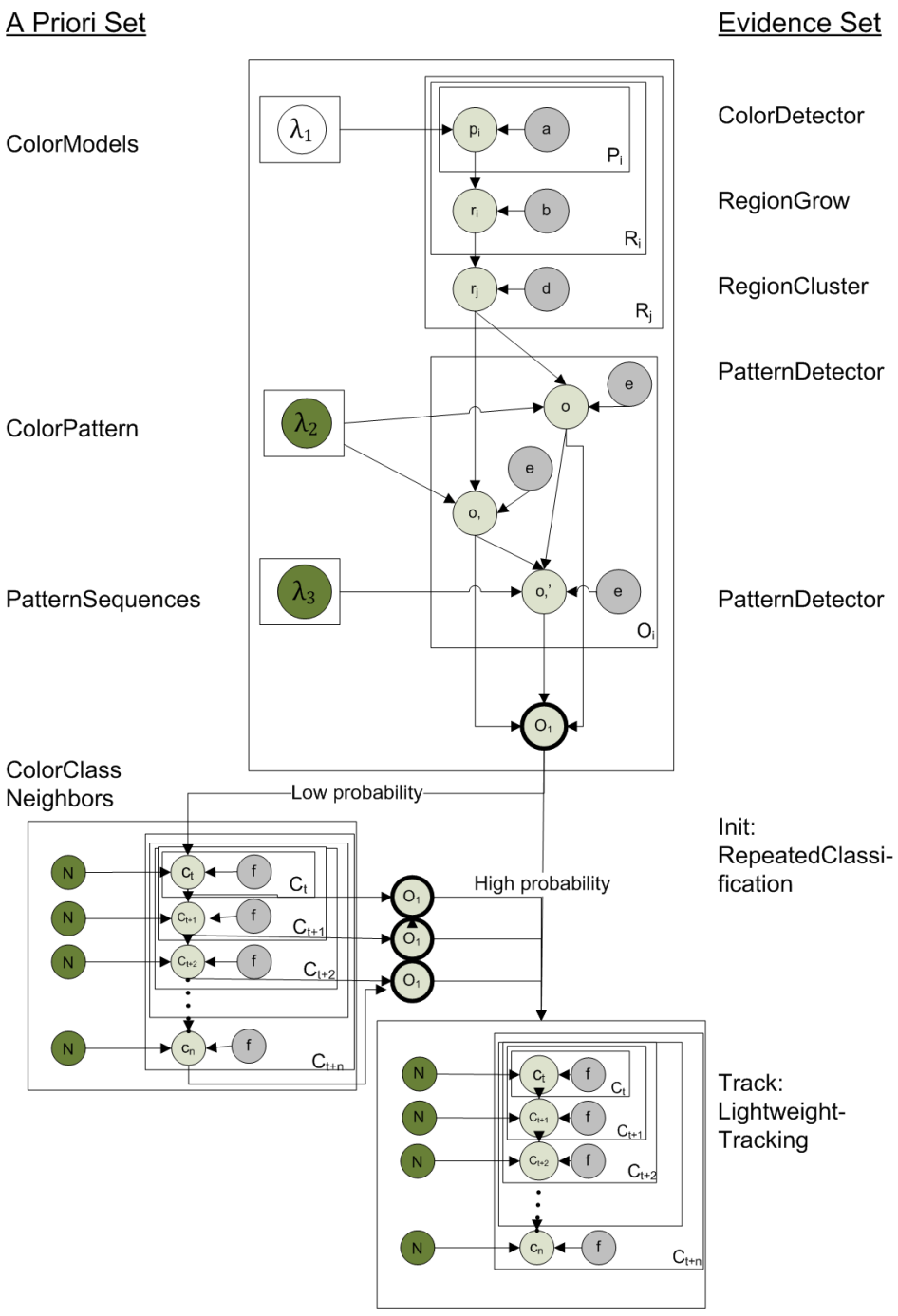

Figure 4.60: Resulting BN concept referring to the box in figure 4.59. Variables: a, b, d, e are detector outcomes, P is a group of pixels (image), R_i and R_j are regions of pixels due to main or pattern color, o and $o^{'}$ refer to main and pattern color, likewise $o_{,}^{'}$ describes the patterned object. $_{1,2,3}$ model the distributions of a priori information.

time steps. More details to the single BNs are given in section 4.4.3. The following sections are used to explain the modeling of the prior information.

4.4.2.1 Bayesian Networks versus Markov Nets

Markov nets (MN) are undirected graphical models. With undirected structures, only the relevance between nodes is representable, not the causal relationships. While BNs have the possibility to map non-monotonic dependencies, the MNs can handle cyclic relations. But with respect to probabilities, only BNs provide mechanisms for inference. On the other hand, in BNs much effort is necessary to map the nodes in semantically correct order. BNs fulfill the Markov property by definition, as nodes depend on their parents. This property leads to the possibility to transform both kind of structures into the other. In this context, the approach deals with a successive detection of a priori information, which leads the choice to the marginal, conditional probabilities, represented in the ordered structure of BNs, even if MN are more practical and easier to construct [110], [99], [23].

4.4.2.2 Color Management

To understand digital models of color, the knowledge of the underlying *radiometry* is helpful. Light is an energy, that light sources emit onto object surfaces. To describe a physical model of this, the measurement of the outgoing energy and the behavior on surfaces is necessary. A classical model of light source position and ray directions, measured as *radiance*, is the *bidirectional reflectance distribution function* (BRDF)

$$p_{bd}\left(\Theta_o, \Phi_o, \Theta_i, \Phi_i\right) = \frac{L_o\left(P, \Theta_o, \Phi_o\right)}{L_i\left(P, \Theta_i, \Phi_i\right) \cos\Phi_i \sin\Theta d\Theta d\Phi} \tag{4.16}$$

which models "the ratio of the radiance in the outgoing direction to the incident irradiance "[5]. In the equation, $L_o\left(P, \Theta_o, \Phi_o\right)$ describes the emitted radiance coming at an solid angle $\sin\Theta d\Theta d\Phi$ at angles Θ, Φ and the received irradiance is given by $L_i\left(P, \Theta_i, \Phi_i\right) \cos\Phi_i$ for a patch of the surface. Due to the fact, that light can be split into wavelengths, the radiance is generally written as $L^{\lambda}\left(P, \Theta, \Phi\right)$ with λ as specific wavelength. Daylight is a combination of a range of wavelengths of different spectral power. The variations of the spectral power are the difficulty in the modeling of a robust model for the color detection of the object detection. The photo receptors of digital cameras have different spectral sensitivities. Generally, the response of a sensor element k due to a surface patch is

$$p_k = \int_{\Lambda} \delta_k\left(\lambda\right) p\left(\lambda\right) E\left(\lambda\right) d\left(\lambda\right), \tag{4.17}$$

with Λ as range and $\delta_k\left(\lambda\right)$ as sensitivity. Standardized models for unified color spaces can be divided into linear and non-linear models. Examples of linear color spaces are *CIE XYZ*, *RGB* (base colors red, green and blue) and *CMYBlack* (base colors cyan, magenta and yellow). Non-linear color spaces are more adequate to human color perception. Representatives of this group are the *HSV* and *CIE LAB*

[5] [63], p.61

space. HSV is the abbreviation of hue, saturation and value as it represents colors independently of the brightness value and is often used to reach color constancy. CIE LAB represents a color space where green and red refer to an axis and equally blue and yellow, whereas L represents the lightness as up-axis. The CIE LAB space tries to model an equal distance for all colors, to get better measurable uniform color differences. The *CIE XYZ* space (or its reduced representation as *XY* space) is nowadays not often used in practice, but refers as theoretical base, whereas RGB and HSV are widely used color spaces.

4.4.2.3 Color Segmentation

An excellent work on the state-of-the-art of image segmentation until 2005 has been compiled by Jaume [218]. The approach for this context has to find color features with prior color class information in outdoor scenes with fast changing light conditions due to light intensity and color temperatures [63]. The approach should be supervised, as unsupervised methods like *Mean Shift* are too costly and there is a priori information, which should be used. A lot of possible approaches have been presented and build the wide field of this topic. For color segmentation approaches the methods can be divided into motion-based, physic-based, edge-based and pixel-based. Edge-based detection in bad outdoor conditions is not robust, therefore the focus is not set on these methods, as well as motion-based methods are too time-consuming.
Very interesting is the segmentation by physical hypothesis [129], but unfortunately not robust enough yet, to deal with the circumstances of an outdoor environment. Image segmentation methods by graphs have in general no real-time performance, but give very detailed results [105], [177]. Beside this, its more costly to first segment all regions of an image and afterwards decide which are interesting due to color descriptions. The faster way is the other way around. In the field of pixel-based methods, some interesting approaches use histograms, that neglect spatial information, and use structural tensors to achieve invariance in rotations and translations [44], but for this large training sets are necessary. Neural and statistical methods are equally powerful, even though the neural nets turn out to be more robust in very complex settings [120]. Good results gives color segmentation with color watershed, combined with Bayesian pixel classification in supervised [118] and unsupervised manner with region adjacency graphs and RedGreen, RedBlue and BlueGreen color pairings [117].
Instead of watershed, region growing can be used, but the problem of region growing is the expensive and sequential character and the challenge of finding adequate seed points. Region methods are often very dependent on the level of detail of the objects. Many approaches exist which model objects in parts and form dependencies between the parts by adjacency graphs, general graphs and trees [158], [176] or statistical spatial correlation [180], [210], [37]. In this context, these approaches will fail, due to the very small objects and blurred regions. An overview of region detectors give [136]. *Gaussian Mixture Models* show a very good performance [164], [123], [223], also in the context of background subtraction [152].
A special topic which has similar requirements like color types and natural envi-

ronments is the field of road sign detection. In [141] a review of the topic is given. They divided the subjects of this topic in color-based approaches, shape-based and other approaches like decision trees and *Nearest Neighbor* classifications. The best performance give methods that are based on shape or color neural networks and nearest neighbor classification which do not work here because of large training or data sets. HSV/HSI or *CIECAM97* color spaces also perform well and are very interesting in this context. Likewise used are feature-based methods like a SURF (Speeded Up Robust Features) approach [147].
The HSV space is very interesting, as the light-independent color information is represented only in the Hue value, and Hue is invariant to gamma variations [61]. An important fact is the chosen color space. A detection of orange rescue targets is given in [200], where the authors subtract the red and green bands from each other. Due to the fact, that this work models various colors, the effort of segmentation would be too high for this purpose. Others use RedGreen, RedBlue and BlueGreen color pairings to segment in different layers, which is very costly and affected by illumination changes [112].
For the modeling of the water surface as background, [165] presents a saturation-to-brightness ratio to describe the specific appearance of water surfaces, with good results, but the question is if other wet surfaces e.g. in the rain, give false positive results. Nevertheless this can be a supporting approach for the color class foreground detection.
Due to given rough a priori color information, real-time requirements and the outdoor character of the lighting conditions, a pixel-based class conditional probability approach in a light independent color space with multiple PDFs is the choice for this context.

4.4.2.3.1 Probabilistic Color Management Due to the heavy light intensity changes of an outdoor environment and the uncontrollable character, the used color space is the HSV space. That means, the color is defined by its hue, saturation and value. As the color definitions for artificial navigational aids are given in the XY space, they can be converted into RGB values by

$$M = \begin{pmatrix} R = X * 2.3706743 & +Y * -0.900040 & +Z * -0.470633 \\ G = X * -0.513885 & +Y * 1.6253036 & +Z * 0.0885814 \\ B = X * 0.0052982 & +Y * -0.014694 & +Z * 1.0093968 \end{pmatrix}, \tag{4.18}$$

which refers to the reference *white E* matrix. The conversion from RGB to HSV is then given by a standard procedure:

$$V = max\,(R, G, B) \tag{4.19}$$

$$S = \begin{cases} \dfrac{V - min\,(R, G, B)}{V}, V > 0 \\ 0, V = 0 \end{cases} \tag{4.20}$$

$$H = \begin{cases} 0 + \dfrac{G-B}{R - min\,(R,G,B)}, R = max\,(R,B,G) \\ 2 + \dfrac{B-R}{G - min\,(R,G,B)}, G = max\,(R,B,G) \\ 4 + \dfrac{R-G}{B - min\,(R,G,B)}, B = max\,(R,B,G) \end{cases} \tag{4.21}$$

Hues range from 0 to 360, in which red has an overflow from 360 to 0. To simplify the computations, the hue range is shifted by 60 values, which results in a not-divided red spectrum. In an image, the detection of the defined colors for navigational aids can be affected by dirty surfaces, sun reflections, color variations, bleached colors and so on. To model the given color definitions due to the hue values, a ternary *Gaussian Mixture Model* (GMM) is used. A mixture model combines linearly independent

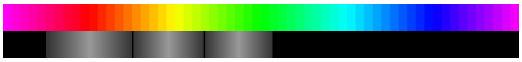

Figure 4.61: Hue spectrum (colored) with probability distributions shown as grey-colored stripes. Dark values belong to low probabilities and bright ones to high values.

probability density functions. The real GMM, detected in a real scene, can vary and therefore, the definitions by the FVT[6] refer as initial guess. The figure 4.61 illustrates the initial densities in the color spectrum. The used GMM models three color classes for the hue representations

$$prob\,(x) = w_r \mathcal{N}\left(x \mid \mu_r \sigma^2_{r,g,b}\right) + w_g \mathcal{N}\left(x \mid \mu_g \sigma^2_g\right) + w_y \mathcal{N}\left(x \mid \mu_y \sigma^2_y\right) \tag{4.22}$$

with each component (red, green, yellow) modeled by an univariate Gaussian distribution

$$\mathcal{N}\left(x \mid \mu, \sigma^2\right) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \tag{4.23}$$

or multivariate Gaussian distributions, which is decided by the observed color class data

$$\mathcal{N}\left(x \mid \mu, \Sigma\right) = \frac{1}{2\pi^{n/2}\left|\Sigma\right|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}. \tag{4.24}$$

In the case of multivariate distributions, the probability is found by MAP

$$\Theta_{MAP} = argmax_\Theta prob\,(\Theta \mid x)\,. \tag{4.25}$$

Due to the conversion into the HSV color space, the estimations are significantly reduced. The values of the GMM due to a measured color are used as a weight for further estimations. The probability of a hue value to be a member of a given color class will be the region classifier for the foreground detection. The model has to be

[6] FVT Koblenz, Wasser- und Schifffahrtsverwaltung des Bundes, Fachstelle der WSV für Verkehrstechniken, Aufsichtfarben für Schifffahrtszeichen, 2010

adapted incrementally in, here called *innovation steps*, by a data fitting approach. Here the ML-based *expectation maximization* (EM) procedure is used. The EM procedure is an iterative algorithm where each iteration finds a better solution due to the ML of the GMM referred to the trainings data. Starting with an initial guess, the EM computes the distribution for the given measurement data. As a correction of the parameters, a second step improves the parameters by the maximization of the joint distribution due to all measurement values. With the adapted parameters the loop over these two steps is repeated until a convergence is reached. These steps allow a stepwise improvement of the parameters. The example detections in figure 4.62 show the detection results for the initial color distribution. As the examples show, the color detection is an oversegmentation and semantic approaches are needed to concentrate the data to the meaningful objects. Due to this problem, the usage of a priori information with the procedure of BNs is introduced in the next section.

Figure 4.62: Color detection results are shown by colored regions in the images; the color specifications of the FVT are used and show an expected oversegmentation because several objects have similar colors.

4.4.3 Bayesian Network Modeling

4.4.3.1 Static Structure-Class Knowledge

This section explains the possibilities of an object detection process fed by a priori information given by sea charts. The main aspect here is to show the usable semantics and topology evaluations which can help to guide detection algorithms through the image processing pipeline. The detection process has to be a cascade structure, which leads to a stepwise detection of shape, color and color patterns. The next paragraph explains the basic structure of the information data base which is deduced by the given sea chart data base.

4.4.3.1.1 Semantics of Navigational Aids The basic information of the object catalogue are given in section 4.4.1.1. Additionally, the definitions for navigational aids by the IALA are given in section 4.4.1.2 with the reference given in the

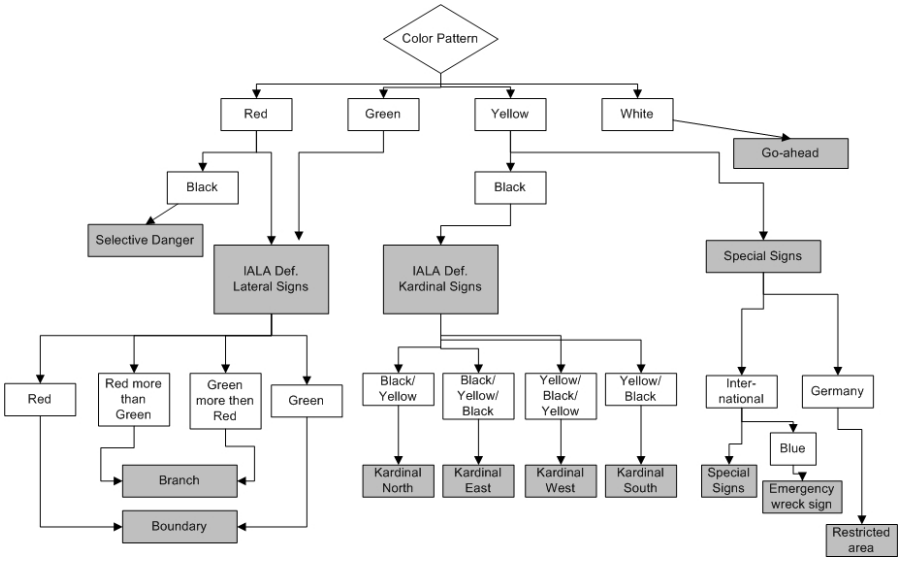

Figure 4.63: Schematic illustration of the semantical color pattern definitions for the body of the navigational aids.

appendix A.1.1. As in section 4.4.1.2 mentioned, the most useful appearance-based information given by the chart are color, color pattern and shape information of a few groups of objects. The depending semantics which can be drawn by these

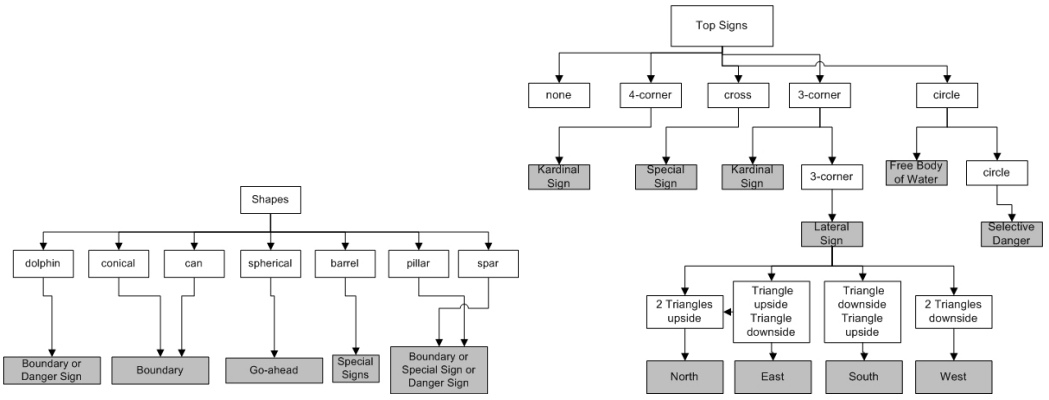

Figure 4.64: Schematic illustrations of the shape of navigational aids and the semantical color definitions for top signs.

definitions show the following figures. Figure 4.63 illustrates the given dependencies and their meaning starting by the color pattern of a navigational aid object. The colors correspond to the body of the navigational aids. For the top signs, the figure 4.64 gives the overview schemata. The figure also shows the main shapes and their meaning. With these information, the structure of a BN can be given.
The random variables can be modeled by different types of nodes: *Chance* like *General* or *Noisy max*, *Deterministic*, *Equation*, *Decision*, *Value* or *Multi Attribute*

Utility or they can form a sub model. Furthermore, for diagnostics it is important, what kind of data type the node is *Target*, *Observation* or *Auxiliary*. The net consists of 59 nodes, 191 states and therefore 37543 parameters. The pure color representatives (targets) are nodes of the type *General*.
There are 21 nodes of this kind, given by the possible color combinations. Special nodes collect the a priori information of possible directions of the patterns, resulting in observation nodes.
To sub models are defined: "color detection" and "buoy classes". The first one models the a priori information due to the given color possibilities and the second one estimates the results of the detections. A sub model named "buoy classes" represents the IALA buoy specification deduced to color and color pattern. To be able to set the evidence for the input node, the sub model "color detection" has to be successful. The specifications of the FVT for usable colors allow to build up the initial guess for a global color model. One node models the actual color model with its color distributions. The nodes feed their observations into collecting nodes which models the probability of a pixel to be an interesting color representative.
Special nodes model the outcomes of the region growing approach and stand for the observations for a colored region.
If there are more than one regions, a color pattern can exist. By projecting the possible objects, given by their positions in the sea charts, the regions can be dedicated to a possible object. If this is the case, the referring nodes will be evidenced.
This color sub model is highly driven by the internal, adaptive color model, the sea chart information and the detection approaches, which fed their results as soft evidence into the probability tables of the nodes. The mathematical representation of the conditional dependencies is given by:

$$\begin{aligned}
&\sum_{n=1}^{N}\sum_{m=1}^{M}\prod_{K=1}^{K} prob\left(x_k \mid pa_k\right) \cong \\
&prob\left(o^{'}, e, o^{'}_{,}, o, n_1, n_2, n_3, n_4, r_i, r_j, a, b, d, e, p_i\right) = \\
&prob\left(o^{'}_{,} \mid e, o^{'}, o, n_2, n_1\right) prob\left(o^{'}, r_j, o, n_3\right) prob\left(o \mid e, r_j, n_3\right) prob\left(r_j \mid r_i, d\right) \\
&prob\left(p_i \mid a, n_4\right)\left(\lambda_1\right)\left(\lambda_2\right)\left(\lambda_3\right)\left(\lambda_4\right)
\end{aligned} \tag{4.26}$$

For explanations of the variables see Figure 4.60. n_1, n_2, n_3 and n_4 refer additionally to helper nodes, which only collect the outcomes of other nodes and K is the number of nodes. Both sub models together form the basic knowledge of sea charts and can be used to predict an object class. They are not able to identify single instances of the classes to calculate their real position and to estimate the users orientation. This BN is a static net with known structure. The evidences are set by likelihoods estimated from the sea charts or by the outcome of the detection approaches for colors, regions, patterns and shape. The outcome of these BN models is a class voting. This can be used to build a second BN, which highly depends on the positions and the predicted object classes. Whether the structure nor the parameters are known in advance, which leads to a dynamically built net during the processing time.

4.4.3.2 Dynamic Structure-Instance Knowledge

This BN sub model can identify single object instances concerning a predicted object class. If the previous detection steps result in a secure prediction, the chart data base can be used to search for possible objects of this class. If more than one object was found, a decision approach has to be used, to predict the correct single object. In this context, if there are more than one object that belong to the same class and that range in similar distances to the user (tilt evaluation), the prediction can not be stated by analyzing a single observation. Due to this, the probable objects have to be saved and under the assumptions of the device's motion, the most probable following objects have to be detected. This information is saved by the dynamic BN which is built when the class information is given. For each probable object, the most probable successors are searched. Because of the possibility of a false detection of the closest follower, the next 5 objects are also modeled by the BN.
To be able to model a voting scheme for the predicted classes, the BN models the dependencies for the best three class predictions. The nodes are uniformly auxiliary nodes, as they wait for the evidence that they have be seen by the detection approaches. The representation is given by

$$\begin{aligned}
&\sum_{n=1}^{N}\sum_{m=1}^{M}\prod_{k=1}^{K} prob\left(x_k \mid pa_k\right) \cong \\
&prob\left(c_n, \alpha_{n_{mr,l}}, \beta_{n_{mr,l}}, \gamma_{n_{mr,l}}, \delta_{n_{mr,l}}, \phi_{n_{mr,l}}\right) = \\
&prob\left(\phi_{n_{mr,l}} \mid \delta_{n_{mr,l},\gamma_{n_{mr,l}},\beta_{n_{mr,l}},\alpha_{n_{mr,l}},c_n}\right) prob\left(\delta_{n_{mr,l}\mid\gamma_{n_{mr,l}},\beta_{n_{mr,l}},\alpha_{n_{mr,l}},c_n}\right) \\
&prob\left(\gamma_{n_{mr,l}} \mid \beta_{n_{mr,l}}, \alpha_{n_{mr,l}}, c_n\right) prob\left(\beta_{n_{mr,l}} \mid \alpha_{n_{mr,l}}, c_n\right) prob\left(\alpha_{n_{mr,l}} \mid c_n\right) prob\left(c_n\right)
\end{aligned} \tag{4.27}$$

with N voted classes, M objects per class and r, l as right and left follower. K is the number of nodes which varies for each net. This is depended on the number of voted classes and found objects per class. Generally, K is in the minimum 6, if only one object could be found. The variables $\alpha, \beta, \gamma, \delta, \phi$ represent the numbers of followers.
In this context, the data can also be interpreted as sequential data. By doing so, approaches like *sliding window*, *hidden Markov models*, *graph transformer networks* or *conditional random fields* could also be used [50]. Due to keep the effort very low and to be consistent in the models, the dynamic BN mentioned above was chosen.

4.4.4 Bayesian Network Processing

The resulting first-level BN structure consists of 15 variables and 60 states, in which the maximum state number for a variable is 13. The aim is to find an effective inference procedure with good performance.

4.4.4.1 Bayesian Network Learning

This section analyses the modeled BNs referring to the question with what kind of techniques the relevant nodes can be updated during the lifetime of the system.

Successful predictions result in correct detected objects which give detailed information about the precision of the used models. The figure 4.65 displays the BN of the first level of the detection process. All relevant nodes with learnable information are marked in green. It follows a description of the techniques and their corresponding context in the used BNs.

4.4.4.1.1 Parameter Learning The learning aspect can have different goals in this context. If the structure is known, only the parameters are unknown and have to be estimated.
The structure of the first-level BN is given by the semantics of the chart data knowledge respective to the specifications of artificial navigational objects. As these information do not vary, the structure is static. The parameters of the nodes are given by different approaches. The first nodes are defined in their a priori parameters by local evaluations of the charts data. It can be more likely to have a sequence of a special color pattern, because the given specification shows this for the actual colors. If only objects on the water surface are searched, the foreground model can exclude colors of coastal landscapes and buildings.
But if the sea chart is evaluated at a specified position, this pattern sequence is more rarely observed than others. In this case, the prior information given by local evaluations are more reliable. These nodes have to update their information depending on the movement in the scene. Other nodes are directly influenced by the image detection algorithms. If there is a learning component within these detectors is an independent aspect here. The information given by the detection stages in the image is newly given by each image in each process level. This color model is used to classify pixel into relevant and not relevant for further detections in the image. The technical specification of the colors varies in real circumstances. The differences will be learned during the procedure of detecting objects which results in a higher detection rate. The blue colored nodes in figure 4.65 are used in this context to verify the concept.

4.4.4.1.2 Color Model The color management was introduced in section 4.4.2.3.1. Due to the decisions made, the color model is represented as a Gaussian mixture model (GMM). During the application the correct detected objects and the objects found after the position update, can give new color information which are collected to update the GMM in temporary innovation steps by EM.

4.4.4.2 Light-Weight Tracking with Bayesian Networks

The observations of the previous BNs can be used to perform a light-weight tracking approach. The classified object serves as starting point for a search in the data base. While it's not known, in which directions the motion of the device will be performed, the neighboring objects of both sides are collected. For each object, the same procedure is repeated. By this approach a tree-like BN is produced.
The possible followers serve as a priori information for the image detection approaches. By projecting the positions of the probable objects the search range in the image is minimized. Additionally, the image processing pipeline can be increased

in their performance, as the appearance-based information like colors and patterns can be specified.
With the advantages of BNs, the classifications can be taken as evidences with their individual classification rate. If a track remains in insecure estimations, a new initialization step has to be fulfilled. Only if the overall probability of the track lies above a threshold, the tracking will be continued.

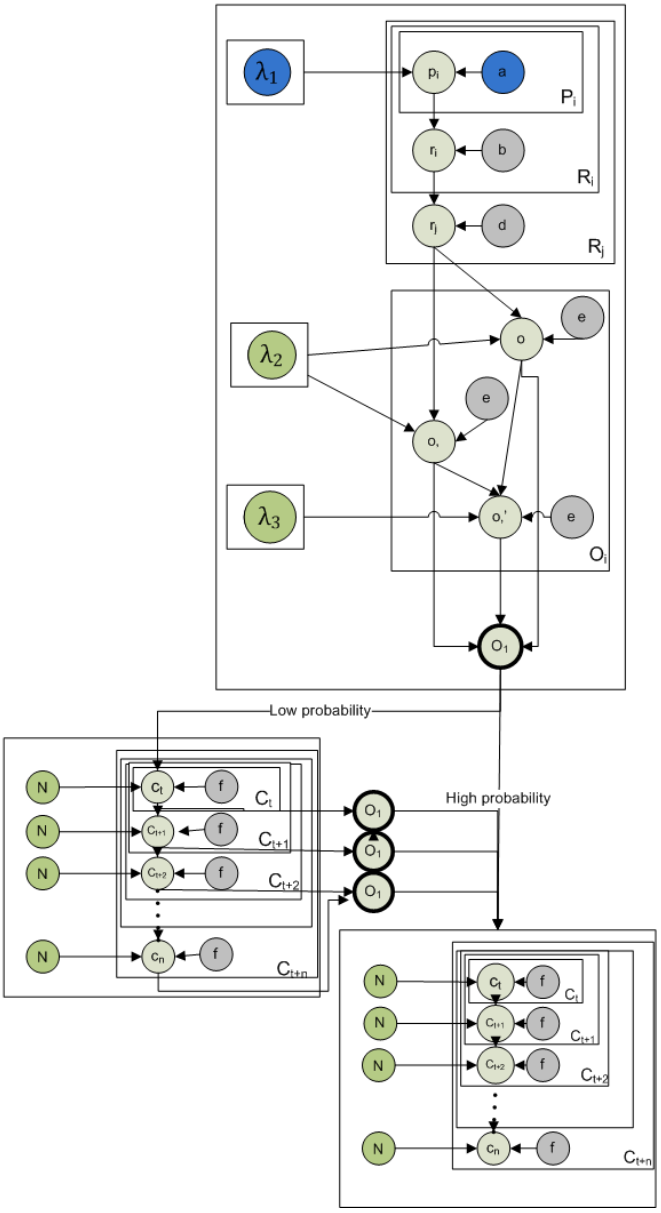

Figure 4.65: Overview of learning and update nodes (in green) and the learning node used in this thesis (blue).

4.4.5 System Evaluation

4.4.5.1 Simulation Data

Due to the fact, that real scenarios for outdoor ARS are too variable and change too fast in their lighting and weather conditions, the first tests have been made on an artificial scene with controlled constraints. The scene was generated as 3D scene with defined artificial navigational aids on a water-like surface. The color variations are modeled with the color catalogue for new and old colors. For a natural context, the objects were positioned in 50 meter, 2 kilometer and 5 kilometer ranges. The figure 4.66 gives some examples of the rendering output. These pictures were used to train the image processing detectors and to evaluate the inference methods for the first-level BN.

Figure 4.66: Examples of the simulation data.

4.4.5.2 Error Sensitivity

The explanations of the developed ARS have shown, that the system consists of several independent measurement units. Together they form the result, which is mainly the augmented view of the user. There are many error types that lead to registration errors between the single units.
In [86], Holloway analyzed the main error types and tried to measure the characteristics. He classified four types: alignment error (error between acquired landmarks and real data), the head/eye-tracking error (due to tracker delays, static and dynamic tracker measurement errors), display error (by optical distortion, miscalibration, aliasing, non-linearity in the display or color aberration) and the viewing error (error between user's location and graphics model). The delay sources alone can be classified into six sub categories, which is the largest error source in an ARS. Holloway found that 1 millisecond delay causes 1 mm registration misalignment. Due to this, the overall systems are characterized by precision, real-time performance, manageability and/or pricing.
Methods of dealing with these errors are approaches like [126] that measure the errors and adapt the visualization by using blurred label positions. By [12] the static errors (like optical distortion, mechanical misalignments or wrong field of view or interpupillary distance) can be measured by continuous calibrations. In contrast, the dynamic errors are inevitable as each unit needs some execution time. They can be minimized and evaluated or guessed, but not avoided. The overall modeling of all

errors in an ARS is not solved adequately yet. Due to this, the overall evaluations measure the frame rate, the overall precision and delay.

4.4.5.3 Evaluation Bayesian Network Concept

The evaluation of the approach is done in different steps. The BNs are structured in a cascade, which refers to correlated effects between them. Due to this, the evaluation is divided into meaningful parts. At first, the first-level BN is evaluated by synthetic and real data. This shows the robustness of the class knowledge, that is then used to feed the second BN for the instance knowledge. The first evaluation is used to find the best parameters in the first-level net.

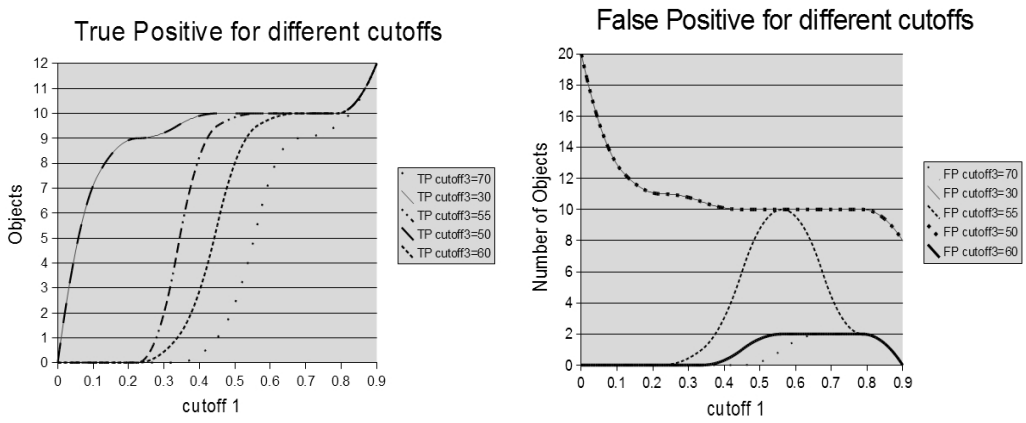

Figure 4.67: True and false positive rates by variations of the cutoff 1 (belief in a priori information) and cutoff 3 parameter (belief referring to the outcomes) with cutoff 2 parameter (evidence propagation) of 1.0.

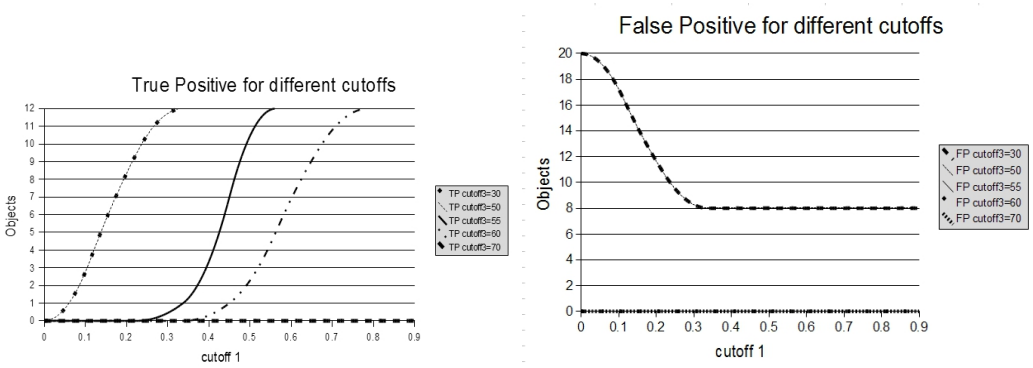

Figure 4.68: True and false positive rates by variations of the cutoff 1 (belief in a priori information) and cutoff 3 (belief referring to the outcomes) parameter with cutoff 2 parameter (evidence propagation) of 0.8.

4.4.5.3.1 Synthetic Data Scene The first-level BN classifies appearance-based information to object classes. The modeling of the structure and a priori information is explained in section 4.4. The parameters for the evidence are detected by

Computer Vision techniques. There are three parameters which influence the performance of the classification. The first one, called *cutoff 1*, is a threshold in the context of the a priori information. It gives a meaning of how sure one can be with these information. The closer the value is to one, the more secure is the information. The value 1.0 is rarely given, as sensors always produce noise and measurement failures. The second parameter, called *cutoff 2*, gives an explanation about the handing-over of values within the BNs. With this parameter, the magnitude of system noise can be modeled. The third parameter, called *cutoff 3*, refers to the belief referring to the decision the BNs made. It models how secure a decision has to be to be a meaningful classification.
The figures 4.67, 4.68 and 4.69 show the variation of the cutoff 1 parameter in relation to selected values for the cutoff 3 parameter (belief referring to the outcomes). The evaluations are done three times, with static but different values for the cutoff 2 parameter (evidence propagation).
The evaluation was done with 20 objects, from which 12 were true objects. The

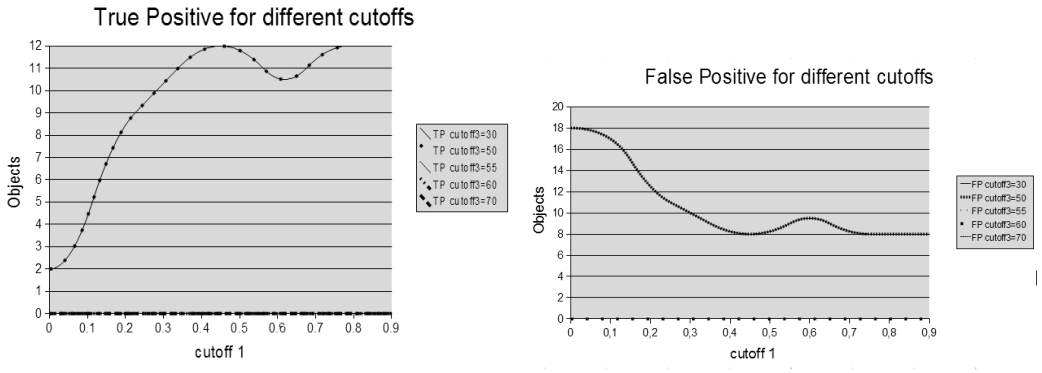

Figure 4.69: True and false positive rates by variations of the cutoff 1 (belief in a priori information) and cutoff 3 (belief referring to the outcomes) parameter with cutoff 2 (evidence propagation) parameter of 0.6.

objects represent 5 of the 21 classes. The instances were colored with the values for new colored objects and with colors of used objects. The figure 4.67 on the left shows the variation of the cutoff 1 parameter. It shows the belief in the a priori information influences the classification very much. Using a cutoff 3 = 70 needs at least a belief of 0.7 to classify 9 of 12 objects correct. The figure on the right shows the false positive rate for the tests. For very low cutoff 3 values in combination with low cutoff 1 values, very much objects are wrong classified. If the cutoff 3 value increases to values above 0.5, the influence of the cutoff 1 parameter rises. This shows the high influence of the cutoff 3 value. The parameter has to be high to minimize the false positive votes. The figures 4.68 and 4.69 present the same tests with lower values of the parameter cutoff 2, which models the evidence propagation within the BN to model the belief in the model. The tests show, that the lower the cutoff 2 parameter, the less useful are the decisions the net makes. If the net is rated to be too noisy, the outcomes are not meaningful anymore. The resulting sensitivity plots can be seen in the figures 4.70 and 4.71. The sensitivity shows the rate of true positive referring to all positive examples and should be high. Although

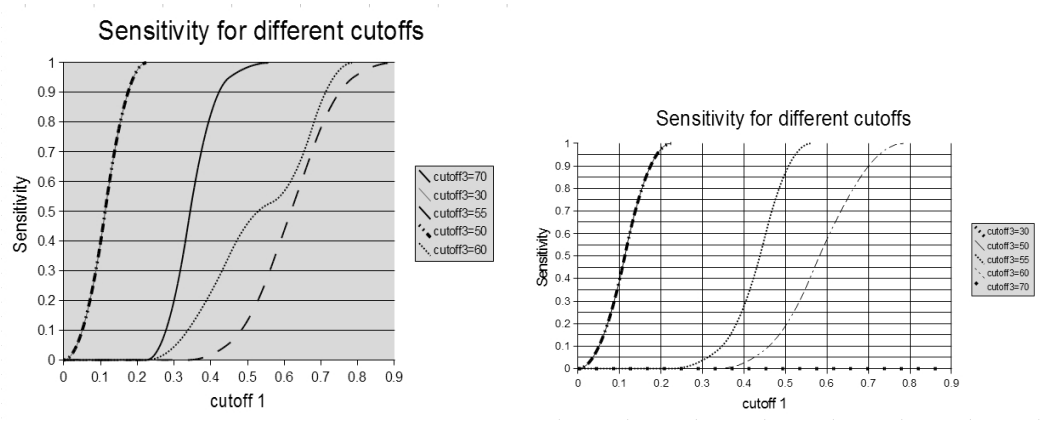

Figure 4.70: Resulting sensitivity plot for left: cutoff 2= 1.0 and right: cutoff 2=0.8. (cutoff 2= evidence propagation)

the BN can be estimated by exact inference methods.
As conclusion, the best parameters, which model the knowledge of the reliability of the sources with respect to the results of the tests, are cutoff 1=0.9, cutoff 2= 0.8 and cutoff 3=0.65.

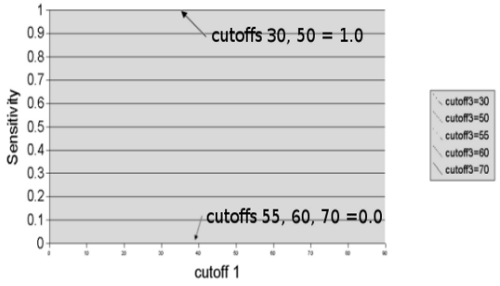

Figure 4.71: Resulting sensitivity plot for cutoff 2= 0.6. (cutoff 2= evidence propagation)

4.4.5.3.2 Real Data Scene The parameters found in the last section are used to evaluate the BN with real data. Given are two real data scenes, one on a very cloudy day and one at sunny conditions, see figures 4.72 and 4.73 for example images of the scenes. The scene with the sunny weather shows very saturated colors, which are easy to detect. In contrast to this, the cloudy sky leads to non-saturated colors, which are not easily discriminable anymore even by the human eyes. The figures 4.74 and 4.75 show the classification performances for class votes and single instances in object series. The class tests were done with the parameters cutoff 2 = 0.8 (evidence propagation) and cutoff 3 = 0.6 (belief referring to the outcomes) over a variation of the cutoff 1 parameter (belief in a priori information). The tests show that the classification of 5 object types are reliable, for all a priori information with a better belief than 0.4. That means that even with noisy a priori information the

classification can be done. The conditions of these results are the reduction of the sea charts data to buoy and beacon classes, reducing the amount of classes to 21. The voted object classes are then used to fill the BN of object instances. The figure 4.75 shows a test, where 4 different series of objects were observed. The series represent the user's scanning of the environment. Usually, the user will start at one object and the next objects will be seen while panning the device left or right. The results in 4.75 show, that there are at least three object observations necessary to find an useful decision. The reason for this is, that in the first detection, the classes are set and all possible instances are added to the BN. In this detection step, no voting for a single object is possible, unless there is only one object referring to the class in the environment.

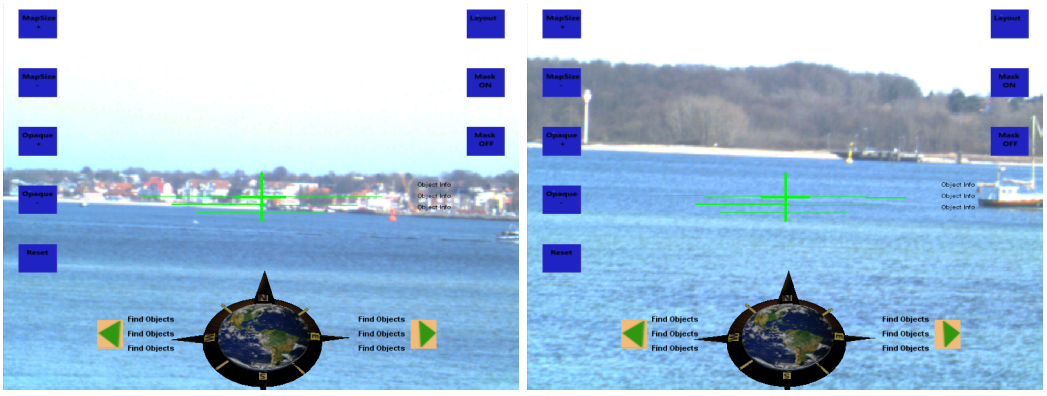

Figure 4.72: Examples of a real scene, sunny day.

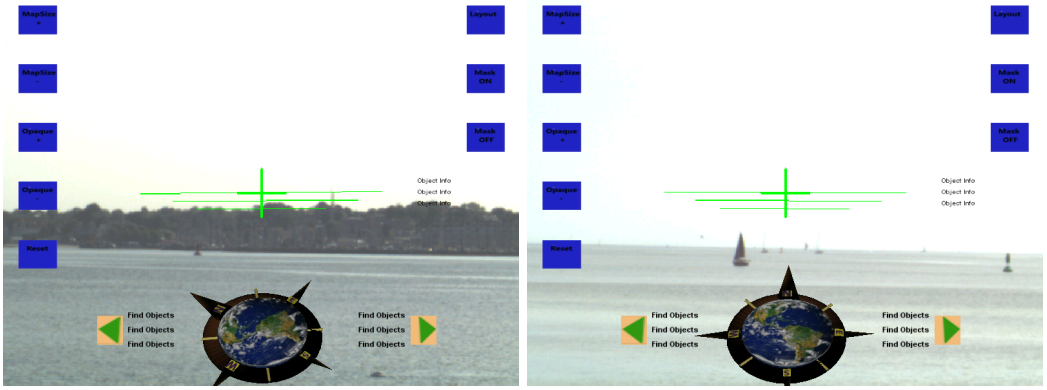

Figure 4.73: Examples of a real scene, cloudy day and backlighting.

As conclusion the more single objects the camera can observe and process the better the classification performance. After the detection of an instance, the estimation of the actual heading is possible and the tracking approach starts.

4.4.5.4 Evaluation Parameter Learning

The BNs can be improved by learning the parameters of the given color model. Different coastal areas use different color models to equip these navigational aids.

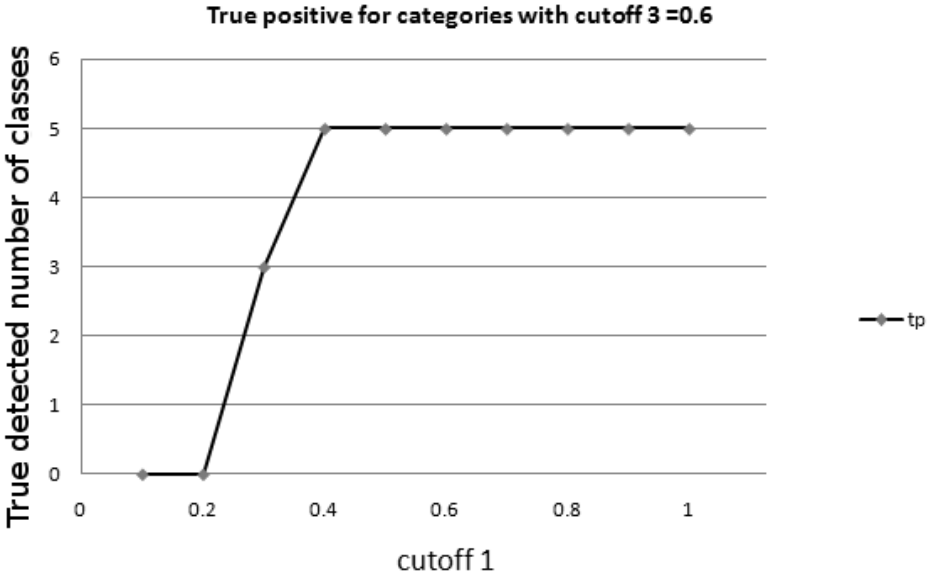

Figure 4.74: Real scene, classification performances: Classes.

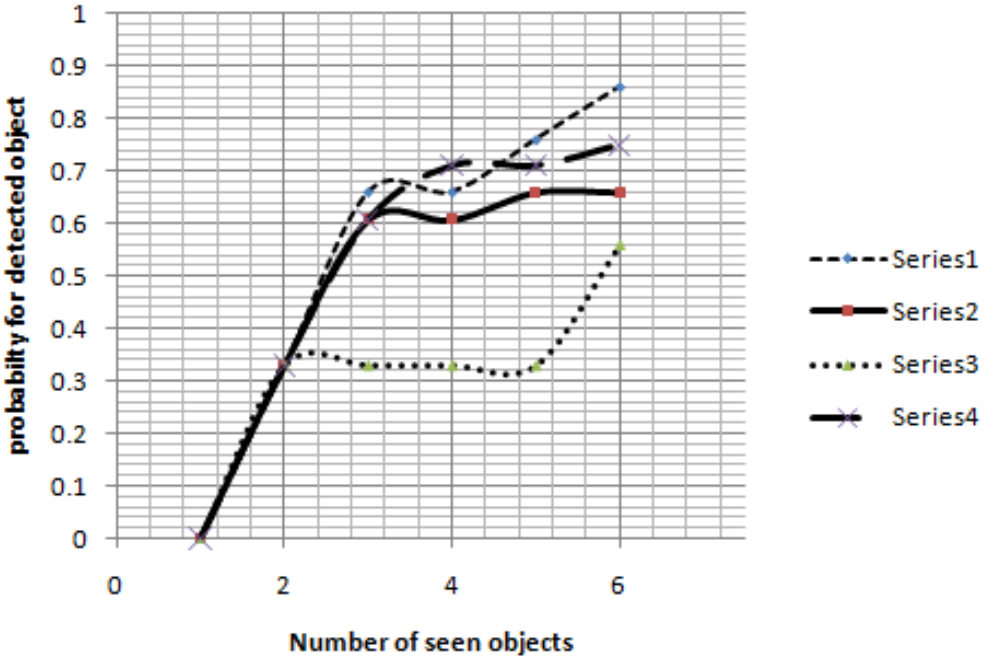

Figure 4.75: Real scene, classification performances: Instances.

In the test area, the special recommendations of the FVT Koblenz hold, but the detection results show a more specific model. The original specification is shown as hue color range in figure 4.76. The grey values refer to the set GMM probabilities, the light ones for high probabilities and the dark ones for low probabilities. The figure 4.77 shows the learned new probabilities. All images with detected objects, as well as some rejected where manually an object was detected, were used to calculate the new function parameters by a Gaussian fit procedure (EM-Estimation). The

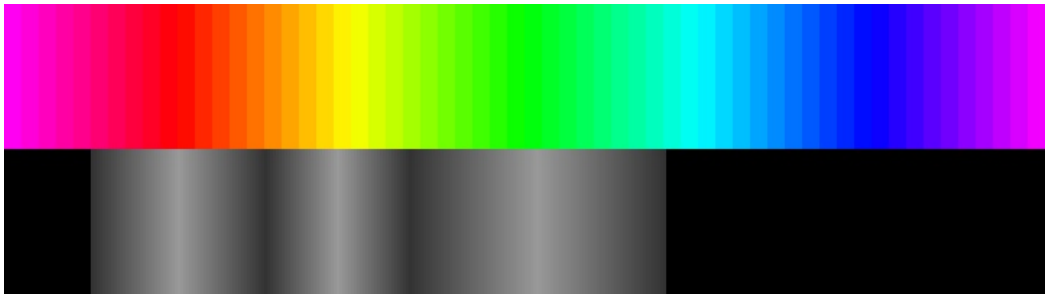

Figure 4.76: Color model without location-specific knowledge; FVT Koblenz color specifications.

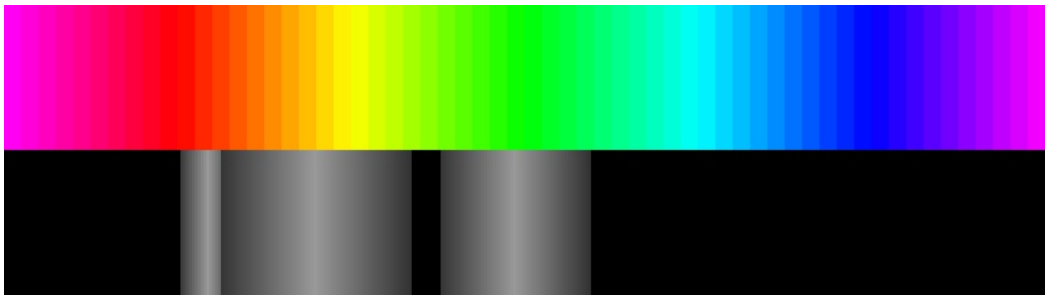

Figure 4.77: Color model learned by positive detection results and test sets.

result shows a much bigger range for the color yellow which refers to lower beliefs and a much narrowed range for the color red which shows a more confident interval. The color green has a better visual fit to the blue colors. The model can be stored and adapted if new coastal areas are visited.

4.5 Overall Evaluation

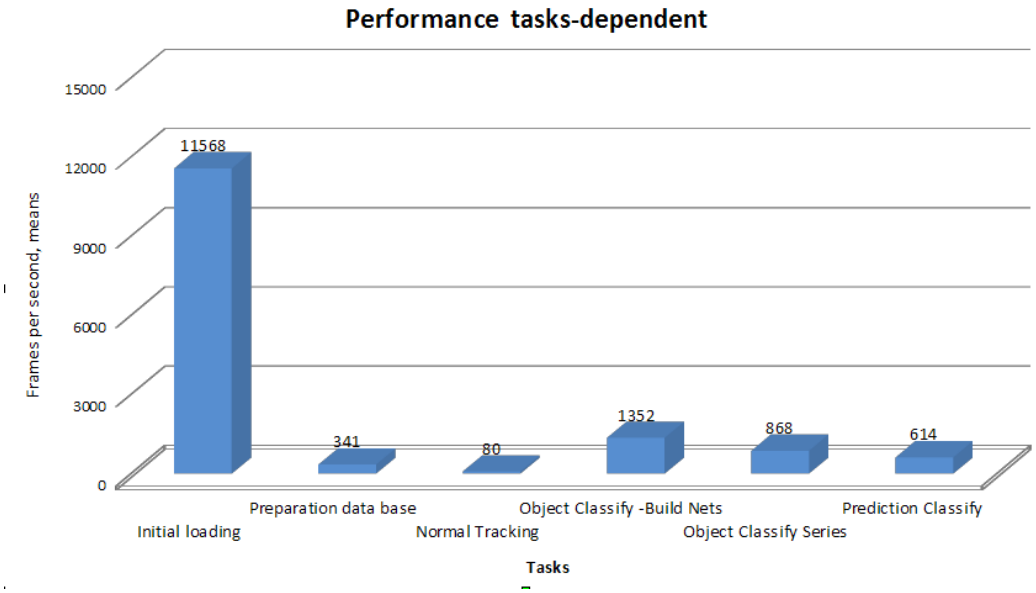

Figure 4.78: Overall process performance per tasks in milliseconds.

To show the overall performance due to the real-time requirements, a number of tests of the time consume have been done. The approach is split into several parts which need different process time.
The application starts with the initial loading of all necessary data, including the most time consuming part of loading the sea chart data base. The time of 11.568 seconds is only needed once for a range of 3 nm.
After the initial loading, the data base has to be scanned due to the needed a priori information. This takes 341 milliseconds. The normal tracking procedure, which contains the pose estimation and the augmentation rendering, needs 80 milliseconds, that refers to 12.5 frames per seconds. If an object is observable, the first object classification is called, which includes the initial building or loading of the BNs.
The structure-known net can be loaded whereas the dynamically built net has to be configured. This initial classification takes in the mean 1.35 seconds.
After this, the following classifications only need 868 milliseconds and if the prediction of objects is used, the process time reduces to 614 milliseconds. That refers to the reduced search range in the images.
To show the overall heading estimation behavior, the precision has been measured. The same test has been done to show the performance of the pose estimation by only inertial sensors, see figure 4.29 in section 4.2. The figure 4.79 illustrates the results for the heading estimation by object detection. Several objects in series have been measured. The heading values are plotted in the figure. With each detection of an object, the heading error is set to almost zero. Remaining inaccuracies refer to delays in the sensor measurements and are negligible concerning the focus of the thesis. In contrast to the continuously resetting of the errors, the increasing gyro-

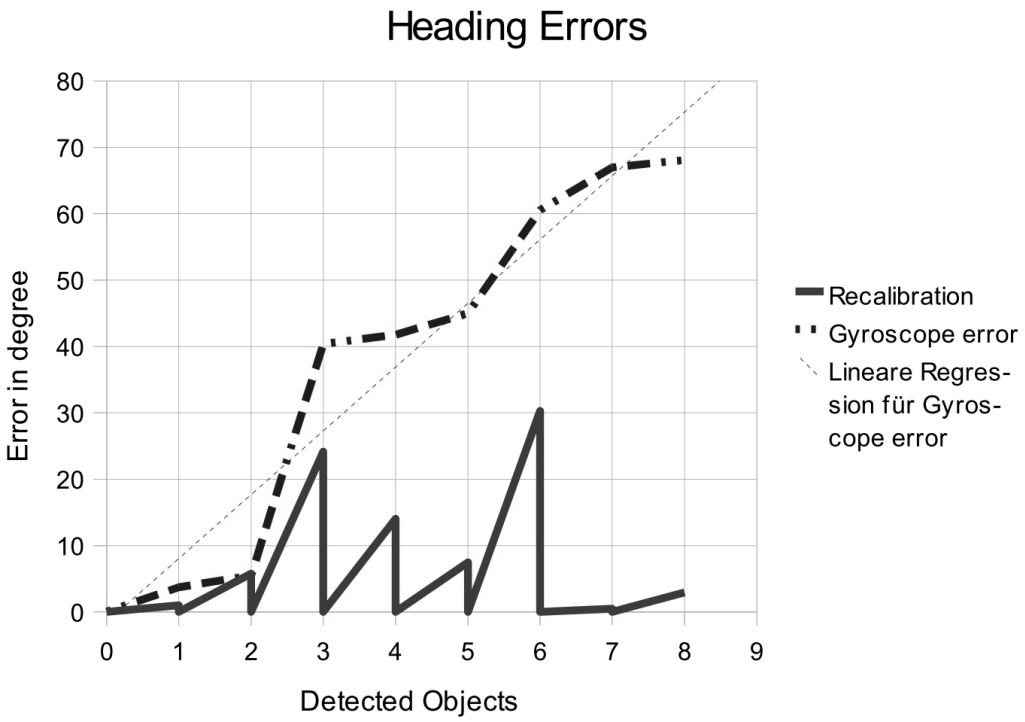

Figure 4.79: Overall test for the heading errors.

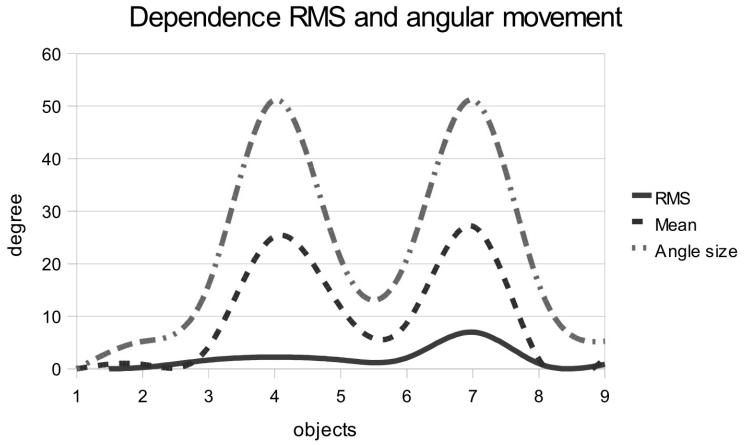

Figure 4.80: Heading errors in correlation to the angular movements.

scope values and a linear trend line for them are shown. The figure 4.80 explains the correlation between the range of angular movement between the objects and the size of errors. The bigger the range between the objects, the higher the errors. This is explainable with the time-dependent drift of the gyroscopes. In this approach, the errors of the gyroscopes does not affect the accuracy, as the full object information can be received each time after the detection. The table 4.82 contains the associated values. The figure 4.81 shows the remaining heading errors after the recalibration. These inaccuracies refer to summed delays between the sensor measurements and show a mean of 0.16 degree. This value refers to 9.6 pixels which is adequate enough to achieve a full overlay of the 3D symbols over the real object in the view.

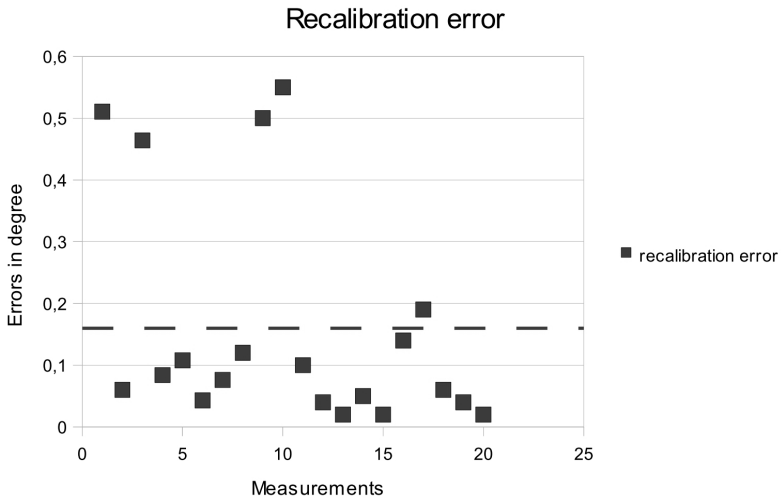

Figure 4.81: Estimation of recalibration errors due to delays in degrees.

Objects	True Heading [°]	Mean Heading Errors [°]	RMS [°]
1st	37.5	0.0	0.0
2nd	42.26	0.85	0.87
3rd	58.79	4.24	1.67
4th	110.0	25.21	2.23
5th	130.9	11.66	1.7
6th	110.0	8.71	2.09
7th	58.79	27.15	6.98
8th	42.26	1.84	1.01
9th	37.5	1.91	0.84

Figure 4.82: RMS estimation of errors in the heading estimation in degrees.

As conclusion, angular ranges of more than 15 degree lead to bigger drift errors. But even with these increasing errors, the object detection approach presented here

is not affected and performs well. The objects are detected, the object description can be received, the heading is recalibrated and a tracking approach can start. This can be repeated each time an object is observable in the video image. Until now, the tracking of objects only refers to the gyroscopes. If the tracking would be supported by standard image-based tracking approaches, the misalignment in the augmentation will be much more minimized and augmentation precisions up to 1 pixel are very expectable as long as the registered object is visible or the next object is identified and can be tracked next.

4.5.1 Visualization

The main aspect of an augmentation device is the information presentation. The pose estimation is the challenging part, but it is only the precondition for an immersive augmentation. The following figures show the resulting hardware design study and the user's display quality. In figure 4.83 the hardware concept is shown in it's practical usage. The touch panel device can be seen with the integrated sensor system.

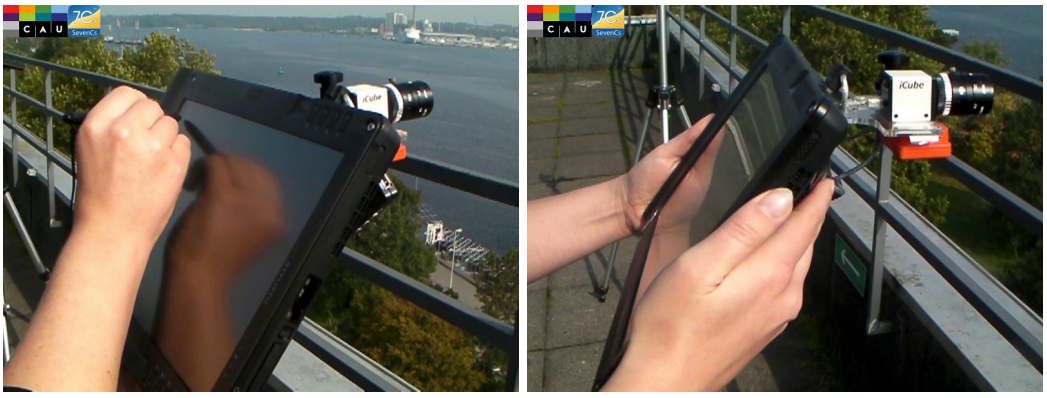

Figure 4.83: Resulting hardware design; Left: Touch panel and integrated cameras. Right: Camera and sensor system at the touch panel.

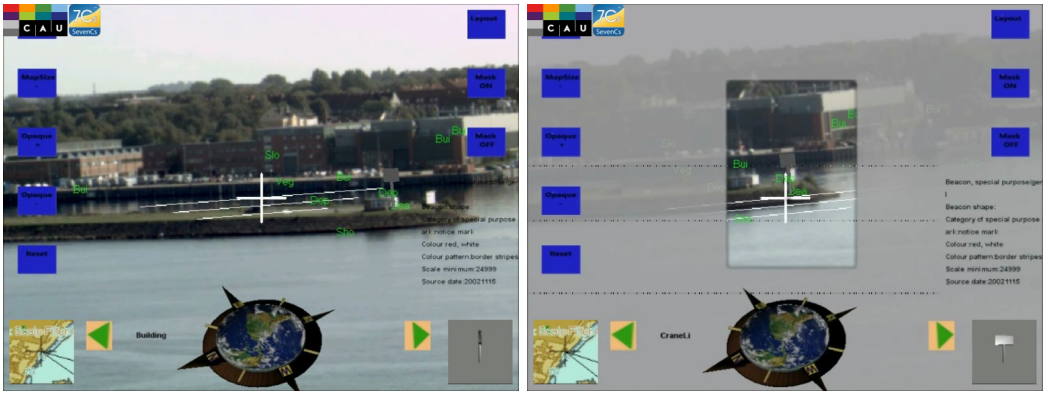

Figure 4.84: Augmentation example. Left: The signal on the right is overlayed. Right: To visualize the focus, a mask can be set.

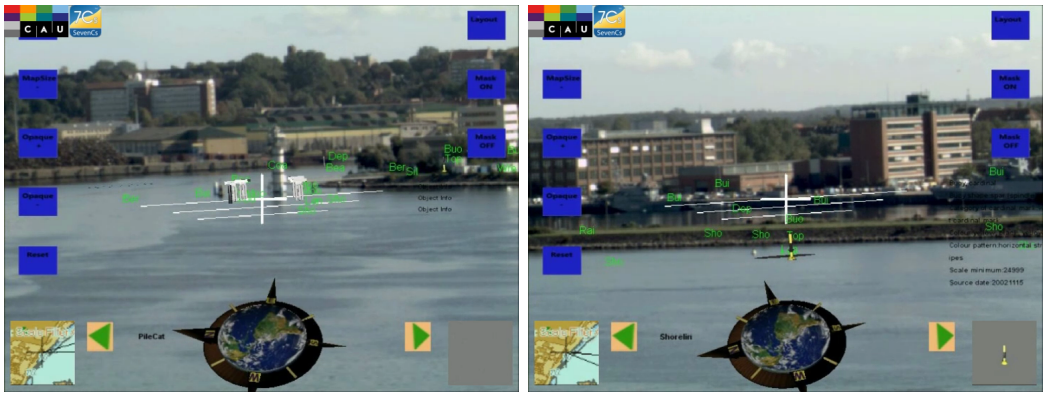

Figure 4.85: Augmentation precision. Left: The misalignment refers to an error of 0.7 degree. Right: Here the error is 1 degree.

Figure 4.86: Examples for missing scaling factor: Left: House symbol. Right: Manually adapted church size that still does not fit.

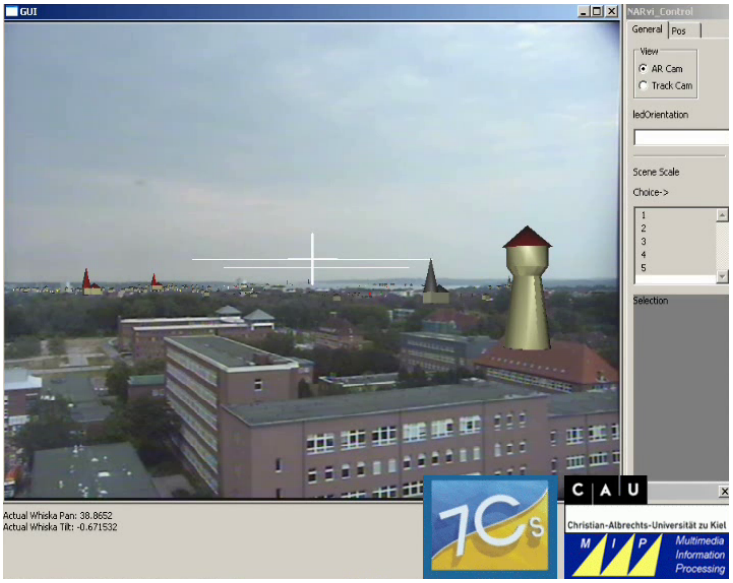

Figure 4.87: The problem with augmentations is the missing immersion if the scene is just overlayed onto foreground objects. The new scene structuring allows the user to set the data focus and thereby the user can deactivate scene parts so that the overlay onto foreground objects can be avoided.

In the figure 4.84 the user's view is presented. The images show the interface design with the main information displays. The background scene is the overlayed image of the environment. On the bottom the user can get navigational information by the moving compass and the 2D sea chart on the left. The small context on the right is activated if a 3D symbol is focused by the user. In this case, the symbol will rotate in this isolated view. Between the contents in the bottom row, the next objects in the left and right direction are listed by their acronyms. In the middle on the right, detailed information of objects can be displayed. If an object is detected or manually focused by pointing with the pen of the device, a mask can be activated to guide the focus more precisely to the object. If the pose can not be estimated precisely, a misalignment will be visible in the images. Examples are given in the figure 4.85. The misalignment refers to real valued errors in the pose estimation. The misalignments refer in their size to the measured errors in the performance tests.
An open issue is the missing scaling factor in the sea chart. Only very few object descriptions contain information due to the size of the objects or at least the vertical length. Due to this, the 3D symbols, which are given in a standard size can not be set without errors. This affects the augmentation precision very much, as the goal should be to exactly overlay the real objects with the 3D symbols. To reach an immersive augmentation, the symbols have to be scaled individually until now. Examples for the lack of immersion can be seen in figure 4.86. At last, the new scene structuring is not only helping the classification approach, it is a big factor to be more immersive. The user is now able to select interesting areas and to hide

Figure 4.88: The new scene structuring allows the user to set the data focus and thereby the user can deactivate scene parts so that the overlay onto foreground objects can be avoided.

uninteresting ones. The advantages were also explained in section 4.3. The user can also change the Level of Detail due to special areas whereas all others stay e.g. in an overview-based detail level. The figures 4.87 and 4.88 show the negative and the new positive examples.

Chapter 5

Conclusion

Contents

5.1 Maritime Augmented Reality

5.1.1 Requirements

To reach a real-time outdoor augmentation device with such a precise augmentation that a magnification factor of 6-7 can be used for wide range objects, several requirements were set. For maritime applications the magnification factor is crucial as it allows the observation of objects at distances of several hundred meters. The challenge in the maritime context is the estimation of the user's heading as a compass is erroneous in the ship's environment. This leads to new sensor combinations. Another aspect is the usage of sea chart as data base for the enhanced view. The standardized sea charts consist of 2D drawing specifications and although there exists a 3D module of the used ECDIS system, the structuring of the virtual scene does not fit to the augmentation tasks.
Due to the set requirements, the topics of the work were split into

- The development of an AR system for maritime application.
- The analysis of 3D sea charts for AR applications.
- The optimization of 3D sea charts for AR applications.
- The development of an approach for precise pose estimation in 3D sea charts with Bayesian networks.

5.1.2 Fulfilled Requirements

The first third of chapter 4 presents an augmentation system that can be used on a ship's bridge with high precision. The system has been verified and patented. The concept fullfils the requirements very well. Different sensors have been evaluated and the combination of inertial sensors for pitch and roll estimation plus the marker-based approach give the best results. The drawback of this marker-based solution is the restricted range for the user movements, as the marker always has to stay in the camera's view.
To overcome this, the marker could be multiplied in the environment or a markerless approach can be used. The second and third part of chapter 4 present the steps to develop a markerless approach.
At first the sea charts are analyzed and a topology-building approach has been defined. And secondly, the usage of a probabilistic object detection approach to estimate the pose is presented. The new data structure allows the user additionally to interact with the scene in an intuitive way. The approach can be practically used and gives an alternative for outdoor pose estimation where compasses are not usable. As long as the weather is not too bad and back light can be avoided to achieve saturated colors, the application can detect artificial navigational aids very well. The identification of instances needs more than one object observation, but this is applicable in a practical solution. The RMS errors of the heading estimations show, that objects with an angular difference of almost 3 degrees have an augmentation error of 0.87 degrees which refers to 52 pixels. Objects with an angular difference of more than 15 degrees have an higher augmentation error which refers to over hundred pixels. But these misalignments are corrected continuously and they will be reduced to almost zero. The remaining inaccuracies are correlated with the delays within the system and refer in the mean to 0.16 degrees which results in an augmentation error of 9.6 pixels. If image-based tracking algorithms are additionally used to support the estimation in addition to the gyroscopes the error between the recalibration can be reduced. The approach can not detect color patterns of objects, if the objects are far way. The camera resolution is the limiting factor in this context. To reach real-time performance, the algorithms have to be optimized. Due to this the evaluation of sampling algorithms is mentioned in the outlook. Although the most challenging topics to build a practical concept have been solved, the most lacking aspect is still the visualization of the 3D scene content. The 3D symbol library for the sea charts that are given to verify the concept, uses very simple symbols. To remain an immersive scene the symbols should match the real objects in their appearance. A problem in this context is the scaling factor of the symbols, based on the missing information of vertical lengths of objects. Only approximations can be used here to date. The new topology helps to allow the focus-based display of objects and the offer of interactivity to the user, but the optical mismatch due to suiting symbols stays.
As overall conclusion, the concept is a very good to identify single navigational objects. Combined with a more precise tracking approach, the pose estimation can be done very precise and due to that an immersive augmentation will be realizable very well.

5.2 Outlook

In the thesis the focus was set to navigational topics. Further possibilities to concentrate on are the detection of natural aids, the visualization and the usage of sampling techniques for inference. Some ideas are described in the following.

5.2.1 Feature Learning for Natural Navigational Aids

The object detection approach introduced in this work uses the appearance-based information of sea charts, which unfortunately is only given for artificial navigational aids.
In some coastal areas it is not adequate to use artificial sea signs due to the sparse appearances. In these cases, it could be more efficient to use the plurality of all navigational aids. The detection approach of this work can be seen as the first step of a learning process.
The classification into known artificial navigational aids refers to a recalibration of the ARS. If this is done, the precise pose of the system can be used to mark the image parts where further navigational aids are expected, due to their geographical position. These image parts can be collected and analyzed to find patterns, recurring colors and other appearance-based information. The very interesting aspect of the detection procedure is, that it can be used to learn the objects that can not be observed by appearance-based information in the chart knowledge base. These objects, like natural features or buildings, vary massively in their object classes, so that general descriptions can not be found and are not given by the charts.
For the learning strategy of this topic, a wide range of approaches are available as this topic is the classical unsupervised image learning task with incremental data sets.
If a heading estimation could be done by the detection of an object, the information given by the data base of the likely positions of the other unobserved objects can be used to define the image area in which the objects are. In these image areas, features like SIFT, GLOH or others can be searched and mapped to a feature space. The classification can be done by classification trees or to be consistent, this can also be done by a Bayesian approach [94].
Unsupervised classification approaches can then be used to find the most likely representatives of the object classes or further more, single instances of the class. [189] estimates the net structure and the net parameters to build an automatic training of detectors for faces and eyes. The statistical dependencies within the relevant image patches are modeled by a BN, which gives very good results. Another Bayesian feature learning procedure can be found in [149], as the authors combine features on which they sample likely new features to replace large data sets. With the help of these features, a BN can be build to discriminate the objects.
In [190] an interesting approach is given that combines local and global feature descriptions by SVM fields. The performance must be optimized for a context like mentioned here, but the robustness is the leading aspect. A very good work on statistical part-based models is given in [235], and could be used to find further solutions for this very interesting topic. By integrating these information into the sea

charts, more applications could use this functionality. With these approaches, all navigational aids can be automatically described and detected by cameras. Additionally, the descriptions could be specified for different coastal areas, e.g. different countries, to be more precise in the color patterns. Furthermore, this approach can be used to verify the content of the sea charts by real-time measurements, by the evaluation of the images.

5.2.2 Visualization

A drawback of the 3D ENC module is the simple structure of the symbols, as there are only few surface textures. To improve this, Computer Vision techniques could supply more information like colors and textures [58][209].
A next step could be to rebuild the referenced models in CityGML by its GML3 geometry structure to have full access to all properties. This process can be supported by camera-based devices as mentioned in this work.
The topology built in the sea charts allows the definition of semantic contexts. They have not been used in the augmented application up to now. They should be used to offer the user specific viewing options and to guide the user through foreign areas where optical impressions can guide the actions.

5.2.3 Evaluation of Inference Strategies

The Bayesian networks used explicit inference to estimate the probabilities. The overall performance shows, that the approaches are not real-time applicable.
Sampling algorithms could be used to reach a better performance. It has to be evaluated if the precision will be decreased and if the effect can be neglected due to the benefit of the possibly reached real-time performance.

Appendix A

Appendix

A.1 General

A.1.1 Color Definitions for Navigational Aids

Die Eckpunkte der zulässigen Bereiche sind:

Tabelle 1: Farbbereiche und Leuchtdichtefaktoren für Aufsichtfarben im Neuzustand

		Eckpunktkoordinaten der Farbbereiche				Leuchtdichtefaktor β	
		1	2	3	4	Standard-farben	Tagesleucht-farben
Rot	x	0,649	0,602	0,701	-	≥ 0,12	≥ 0,25
	y	0,351	0,348	0,249	-		
Gelb	x	0,479	0,456	0,484	0,512	≥ 0,50	-
	y	0,520	0,494	0,466	0,487		
Grün	x	0,100	0,222	0,300	0,300	≥ 0,12	≥ 0,25
	y	0,831	0,543	0,605	0,693		
Weiß	x	0,295	0,305	0,335	0,325	≥ 0,75	-
	y	0,325	0,315	0,345	0,355		
Schwarz	x	0,260	0,300	0,385	0,345	≤ 0,02	-
	y	0,310	0,270	0,355	0,395		

Figure A.1: Color Definitions new objects.

Tabelle 2: Farbbereiche und Leuchtdichtefaktoren für Aufsichtfarben im Gebrauchszustand

		Eckpunktkoordinaten der Farbbereiche				Leuchtdichtefaktor β	
		1	2	3	4	Standard-farben	Tagesleucht-farben
Rot	x	0,649	0,565	0,674	-	≥ 0,12	≥ 0,25
	y	0,351	0,345	0,236	-		
Gelb	x	0,465	0,427	0,470	0,522	≥ 0,50	-
	y	0,534	0,483	0,440	0,477		
Grün	x	0,003	0,187	0,313	0,313	≥ 0,12	≥ 0,25
	y	0,632	0,452	0,566	0,682		
Weiß	x	0,290	0,300	0,350	0,340	≥ 0,75	-
	y	0,320	0,310	0,360	0,370		
Schwarz	x	0,260	0,300	0,385	0,345	≤ 0,02	-
	y	0,310	0,270	0,355	0,395		

Figure A.2: Color Definitions used objects.

	height	pattern	color1	color2	Topshape	shape
BCNCAR	x	x	x	x		x
BCNISD	x	x	x	x		x
BCNLAT	x	x	x	x		x
BCNSAW	x	x	x	x		x
BCNSPP	x	x	x	x		x
BOYCAR		x	x	x		x
BOYINB		x	x	x		x
BOYISD		x	x	x		x
BOYLAT		x	x	x		x
BOYSAW		x	x	x		x
BOYSPP	x	x	x	x		x
BRIDGE		x	x	x		
BUIREL	x		x			
BUISGL	x	x	x	x		
CAIRNS	x	x	x	x		
CHIMNY	x	x	x	x		
COALNE			x			
CONVYR	x	x	x	x		
CRANES	x	x	x	x		
DAMCON	x	x	x	x		
DAYMAR	x	x	x	x		
DSHAER	x		x			
DUNARE	x		x			
FLASTK	x		x			
FLGSTF	x		x			
FLODOC		x	x	x		
FNCLNE	x	x	x	x		
HULKES		x	x	x		
LIGHTS	x		x			
LITFLT		x	x	x		
LITMOI	x		x			
LITVES		x	x	x		
LNDMRK	x	x	x	x		
MONUMT	x		x			
MORFAC	x	x	x	x		
MSTCON	x	x	x	x		
OFSPLF	x	x	x	x		
PILPNT	x	x	x	x		
PYLONS	x	x	x	x		
RADDOM	x		x			
RETRFL	x	x	x	x		
SBDARE			x			
SILBUI	x	x	x	x		
SILTNK	x	x	x	x		
SILCONS	x	x	x	x		
SLOGRD			x			
SLOTOP			x			
TNKCON	x	x	x	x		
TOPMAR		x	x	x	x	
TOWERS	x	x	x	x		
WIMCON	x		x			
WDDMIL	x		x			

Figure A.3: Appearence-based objects of S-57.

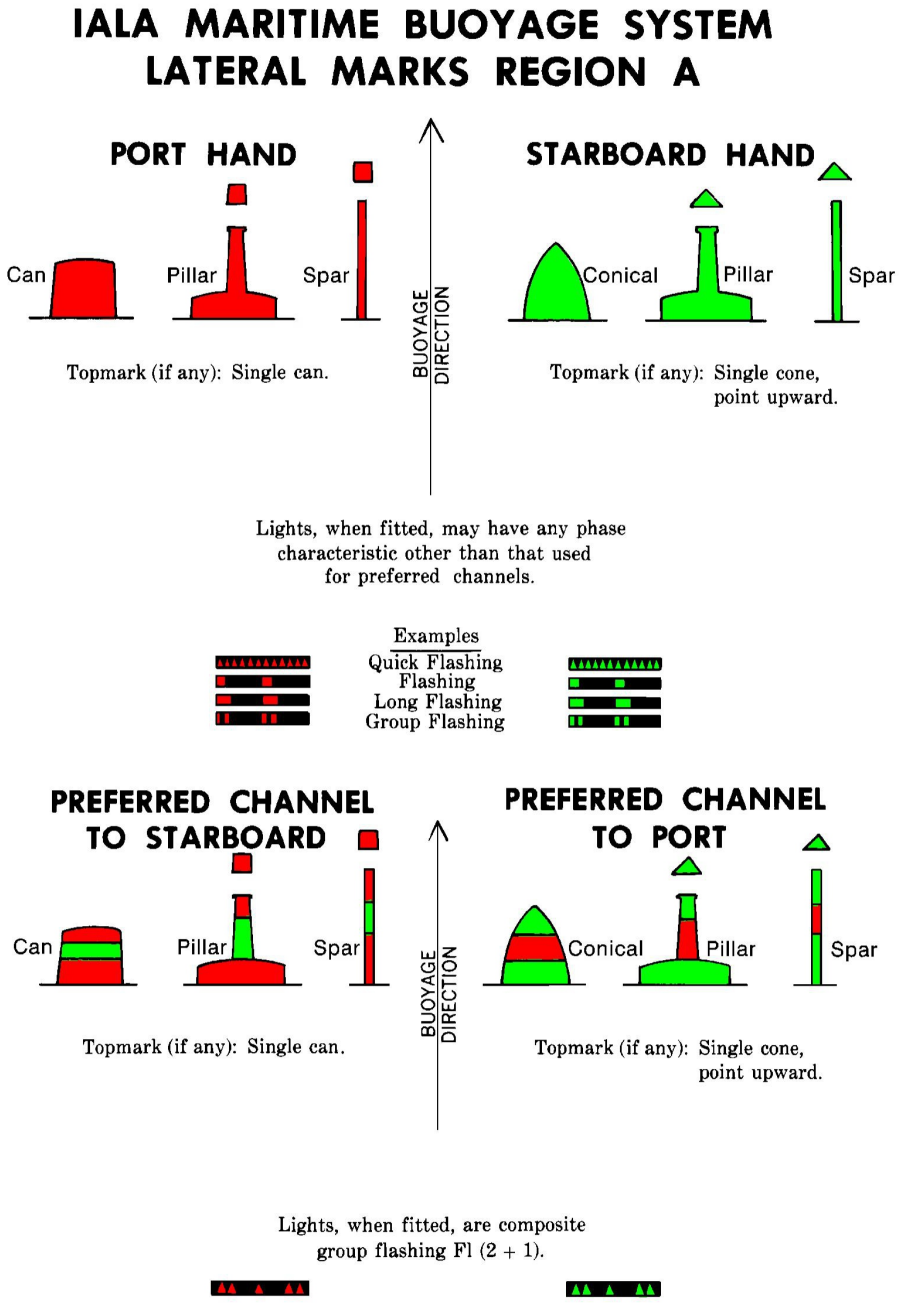

Figure A.4: IALA buoyage system, www.nauticalcharts.noaa.gov/mcd/chart1/, accessed 13.10.2011, page 1.

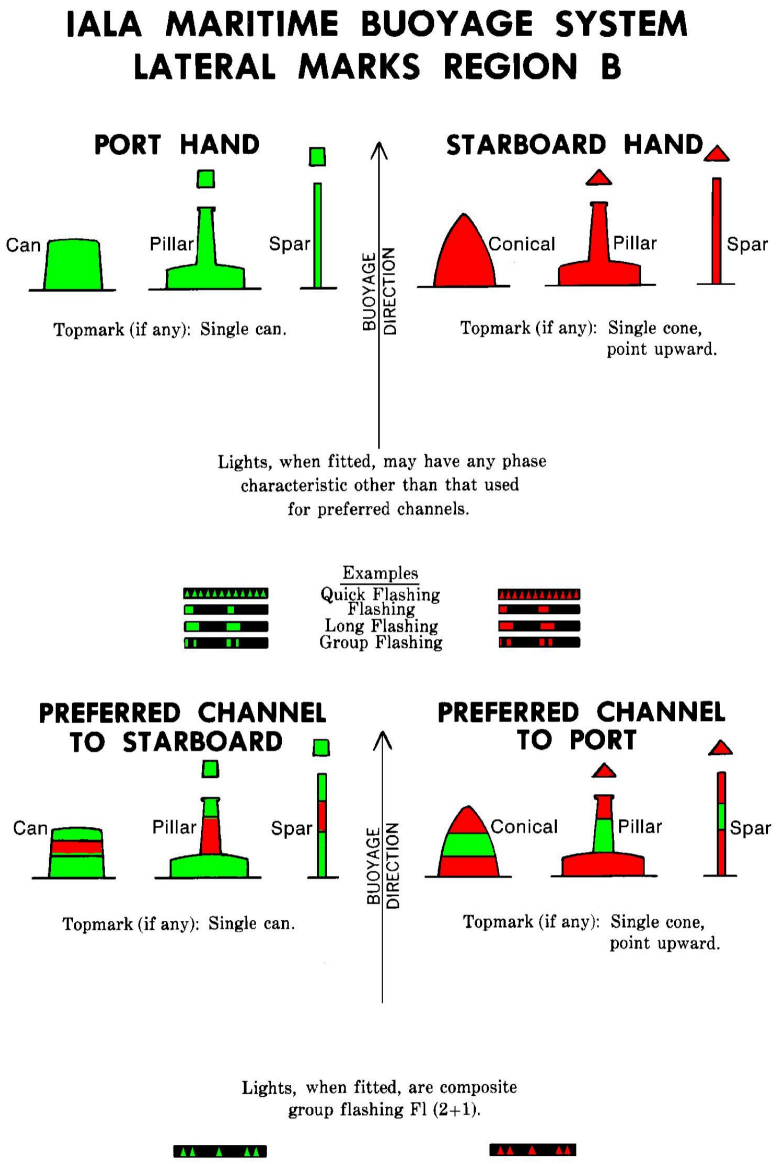

Figure A.5: IALA buoyage system, page 2.

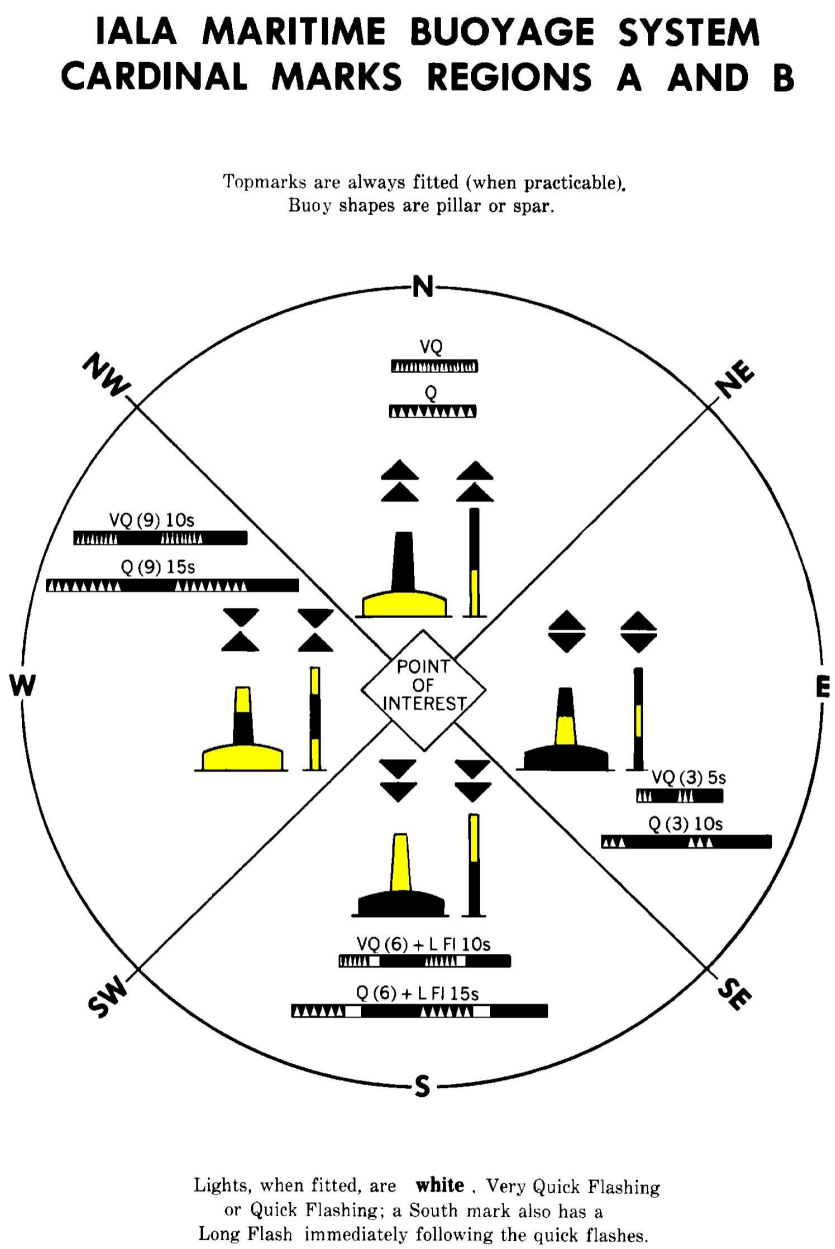

Figure A.6: IALA buoyage system, page 3.

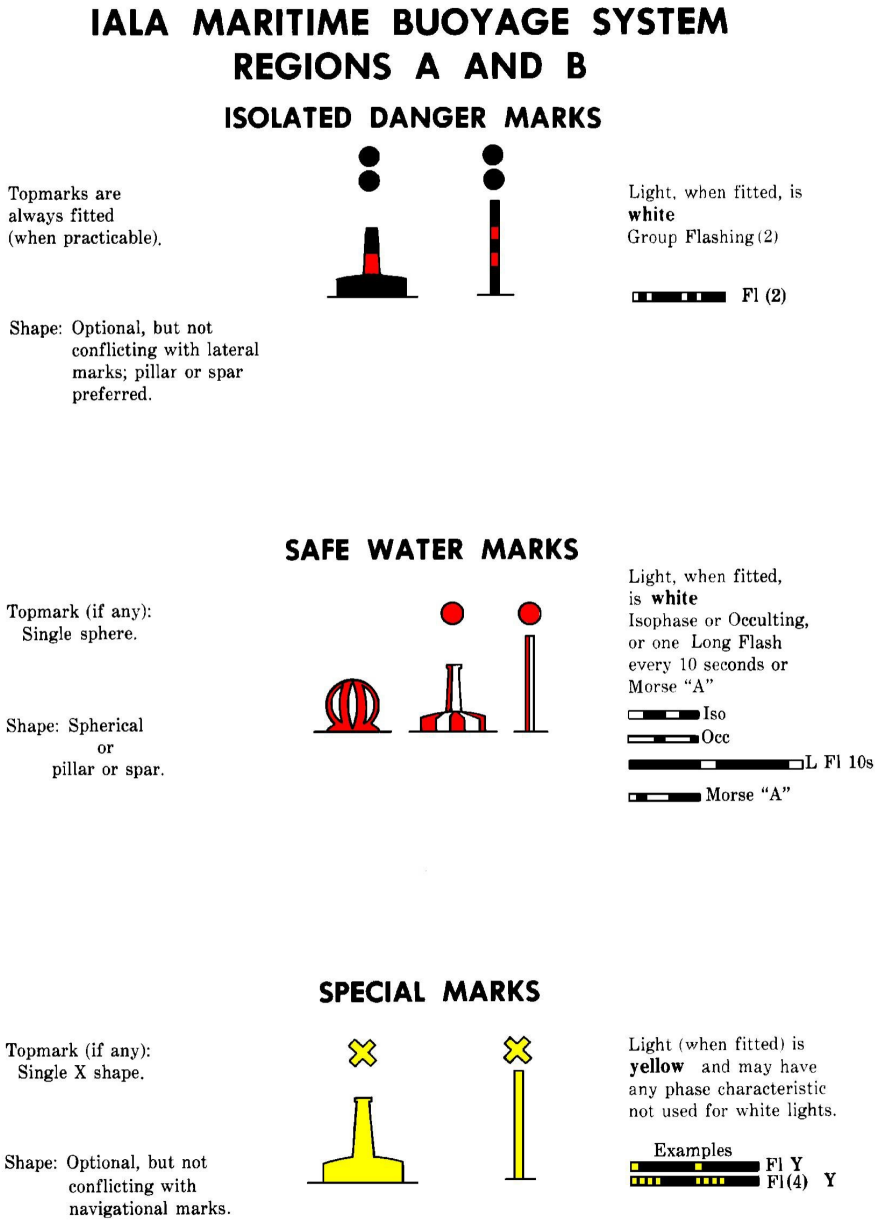

Figure A.7: IALA buoyage system, page 4.

A.1.2 Digital Images

Digital images are captured by a planar image sensor, consistent of MxN, with M,N $\in$ N photosensitive receptors. The mapping is defined by

$$I\left(row, column\right) = \int_{\lambda}\int_{(M,N)} E\left(\left(M,N\right),\lambda\right) R\left(p\right) q\left(\lambda\right) dpd\lambda \tag{A.1}$$

with wavelength λ, and R is the spatial response of the site and q as quantum efficiency which describes the mass of the number of electro-sensitiveness of the chip per incident power. By receiving the light on a surface area, the continuous luminance intensity is captured onto the sensor elements. For a processing in computers, the data is discretized into a predefined digital values range. This quantization for the digital signal D can be defined by the following equation

$$D\left(r,c\right) = \gamma\left(N_I\left(r,c\right) + N_{DC}\left(r,c\right) + N_B\left(r,c\right) + R\left(r,c\right)\right) + Q\left(r,c\right) \tag{A.2}$$

with γ as gain, Q as *Quantization Gaussian noise*. N_I is R modeled with random noise. N_{DC} accounts for the noise by thermal effects and N_B models the bias of defect pixels. The picture resolution and therefore the sharpness is given by the number of sensor elements, while the display quality is given by the dynamic range of the analog signal and in parallel of the range of the discretized values in combination with the used display medium. The frame rate of the capturing is related to the perceptible motion [63].
Due to the problem that human eyes have a non-linear luminance perception and sensor elements capture in a linear way, an exponential function can be used to scale the values more intuitively. The exponent of the function is called *gamma value*, which can be adjusted for every digital capturing device or electronic display. If an object is optimally mapped depends on several factors. The given resolution, the dynamic range, the size of the object influenced by the focal length and camera lenses, fabrication defects, thermal effects, the natural lighting conditions and the kind and amount of image compression are some of them. In cases of motion detection, also the delays in the capturing process are to be considered. Due to these errors, much effort has been done to model the whole process in detail and fix these problems. One topic is the calibration of sensor chip singularities. Digital cameras use the techniques of Charge Coupled Devices (CCD) or Complementary Metal Oxide Semiconductors (CMOS). In [81] is a comparison between these technologies and shows their characteristics. A second topic is the modeling of the geometrical image process, see section 2.1 for a closer look. A third topic is the modeling of the signal processing, as the luminance distribution can be seen as a combination of signals. For the reconstruction of the digitalization process there exist a well-defined mathematical base, see [204] for details. These techniques can also be a medium for image improvement algorithms. The fourth topic is the reconstruction and interpretation of the lighting and color conditions as well as their digital representation. The section 4.4.2.2 will give a brief introduction [97] [233].

A.1.3 Coordinate Systems

Due to different sensors and system modules, there exist a lot of different definitions of coordinates. It is necessary to define these and to find adequate transformations between them, so that measurements can be used in all system components. To concentrate on the main coordinate systems, see figure A.8 for an illustration. Starting with the main point of reference, the earth, described approximately as

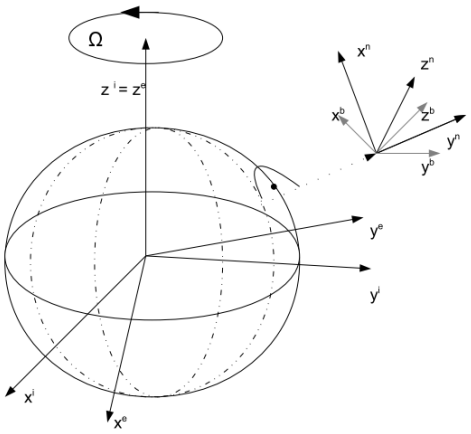

Figure A.8: Schematic coordinate system

ellipsoid with the origin in the center, has an own *Earth-centered Coordinate System* (EarthCS). The positive X axis is pointing to the Greenwich meridian where it hits the equator, while the positive Z axis points to the north pole. Different sensor technologies can approximate the current state of the EarthCS, while defining own coordinate systems of measurement. The IMU, see section 2.1.1.3.1, define the *Inertial Coordinate System* (InertialCS) by using the same origin, but is fixed to the EarthCS and characterized by Newton's law of inertia. The GPS system can use diverse coordinate systems, mostly common is the *WGS84* (*World Geodetic System* from 1984) format which defines the ellipsoid to approximate the geoid. The characteristics are essential to provide preferably exact positions on the earth's surface. It is common standard to define a geocentric coordinate system as tangential plane to the earth's surface. This local system is perpendicular to the earth center but only attaches the earth in one point. In this local reference, the *User-centered Coordinate System* (UserCS) can be defined, which describes the own pose.

A.1.4 Characteristics of Rotations

Rotations are orthogonal, linear transformations $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$, with $x_i = Rx_j$, forming the special *Lie group* $SO(3)$. Generally, a rotation matrix is a 3x3 matrix like

$$R = \begin{pmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{pmatrix} \tag{A.3}$$

with u, v, w as spanning unit vectors. Rotations can have multiple representations with different characteristics (singularities and partial ambiguousness), as follows.

Angle based

In matrix representation, rotation matrices are $n \times n$ matrices with determinant 1, preserving length and accordingly angles, justified by the dot product (invariant under rotations) that can also be defined in terms of length. Rotations have to be defined by context. They can be fixed by an object or point in the coordinate system or fixed at the coordinate system, defined by $x_i = R^{-1}x_j$. Different angle representations are possible, mostly used are Euler angles to decompose a rotation matrix. Euler angles define three angle for three single rotation axis. Due to the normalization of the axis, each has 2 DOFs. A composed rotation matrix has the form

$$\begin{pmatrix} \cos\Psi\cos\Phi & -\cos\Theta\sin\Phi+\sin\Theta\sin\Psi\cos\Phi & \sin\Theta\sin\Phi+\cos\Theta\sin\Psi\cos\Phi \\ \cos\Psi\sin\Phi & +\cos\Theta\cos\Phi+\sin\Psi\sin\Theta\sin\Phi & -\sin\Theta\cos\Phi+\cos\Theta\sin\Psi\sin\Phi \\ -\sin\Psi & \sin\Theta\cos\Psi & \cos\Theta\cos\Psi \end{pmatrix} \tag{A.4}$$

for the *Euler* angles Θ, Ψ, Φ, composed by

$$\begin{pmatrix} cos(\Theta) & 0 & sin(\Theta) \\ 0 & 1 & 0 \\ -sin(\Theta) & 0 & cos(\Theta) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos(\Psi) & -sin(\Psi) \\ 0 & sin(\Psi) & cos(\Psi) \end{pmatrix} \begin{pmatrix} cos(\Phi) & -sin(\Phi) & 0 \\ sin(\Phi) & cos(\Phi) & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{A.5}$$

The problem of Euler angles is the so-called *Gimbal lock*, which describes a degeneration of one DOF by rotating two of three axis into the parallel case. The involved angles can not be estimated anymore.

Axis-angle based

A matrix is symmetric if $A^T = A$ and skew-symmetric if $A^T = -A$. The Rodriguez formula can be derived geometrically and is given by

$$R(\Theta, t) = I + \sin(\Theta) N + (1 - \cos(\Theta)) N^2 \tag{A.6}$$

with angle Θ, axis t and

$$N = skew(n) = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix} \text{ if } n = \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} \text{ [146].} \tag{A.7}$$

Characteristic are singularities for angles close to null. Each rotation is dual defined as $R(\Theta, t) = R(-\Theta, -t)$ holds.

Quaternion based

A unit quaternion is given by

$$q = (v \sin(\Theta/2), \cos(\Theta/2))^T, \tag{A.8}$$

which is defined as rotation about vector v and through angle Θ, $0 < \Theta < 2\pi$. A rotation matrix R_q by quaternions is then written as

$$\frac{1}{q^{2_0}+q^{2_1}+q^{2_2}+q^{2_3}} \begin{pmatrix} q^{2_0}+q^{2_1}+q^{2_2}+q^{2_3} & 2\left(q_1q_2-q_0q_3\right) & 2\left(q_1q_3+q_0q_2\right) \\ 2\left(q_2q_1+q_0q_3\right) & q^{2_0}-q^{2_1}+q^{2_2}-q^{2_3} & 2\left(q_2q_3-q_0q_1\right. \\ 2\left(q_3q_1-q_0q_2\right. & 2\left(q_3q_2+q_0q_1\right. & q^{2_0}-q^{2_1}-q^{2_2}+q^{2_3} \end{pmatrix} \tag{A.9}$$

Equivalent, the axis-angle presentation with quaternions is

$$R_{q'} = I + 2\left(q^2 I + qN_q + N_q^2\right) = \tag{A.10}$$

$$\begin{pmatrix} 1-2\left(q_2^2q_3^2\right) & 2\left(q_1q_2-q_0q_3\right) & 2\left(q_1q_3+q_0q_2\right) \\ 2\left(q_2q_1+q_0q_3\right) & 1-2\left(q_1^2q_3^2\right) & 2\left(q_2q_3-q_0q_1\right) \\ 2\left(q_3q_1-q_0q_2\right) & 2\left(q_3q_2+q_0q_1\right) & 1-2\left(q_1^2q_2^2\right) \end{pmatrix}. \tag{A.11}$$

Quaternions are the most accurate and numerically stable representation, without singularities. Note that the ordering of multiplications is inverse according to matrix representations.

Linearization
A rotation can be linearized by using the quaternion representation and an iteration scheme. The estimation procedure is then

$$R\left(r^{(v+1)}\right) = R\left(r^{(v)}\right) R\left(\Delta r^{(v)}\right) \tag{A.12}$$

where $R\left(r^{(v+1)}\right)$ is a concatenation of matrix multiplications, with small, finite corrections given by Δ. For these small rotations hold $\left[1, \frac{1}{2}\Delta r^T\right]^T$ which leads to $\Delta q^{(v)} = \begin{pmatrix} 1, \\ \frac{1}{2}\Delta r^T \end{pmatrix}$ and $q^{(v+1)} = q^{(v)}\Delta q^{(v)}$ and accordingly $R^{(v+1)} = R\left(q^{(v+1)}\right)$ [131][28][76].

A.2 Statistics

A.2.1 Parameter of Random Variables

In the following there is a list with basic formulations for random variables.

Mean
The *mean* vector is also known as *expected value* and written as

$$m_i = \mathbb{E}\left[X_i\right] = \int_{-\infty}^{\infty} x_i p\left(x_i\right) dx_i. \tag{A.13}$$

Standard Deviation
The *standard deviation* is given by

$$\delta = \sqrt{\mathbb{E}\left[\left(X - \mathbb{E}\left[X_i\right]\right)^2\right]}. \tag{A.14}$$

Entropy
The *entropy* is a mass for describing the number of states of a system by $H(X) = \sum_{i=1}^{k} p_i log_2 \left(\frac{1}{p_i}\right)$.

Covariance
The *covariance* measures the variation of values x_i, x_j correspondingly to the measured mean of the distribution:

$$C_{i,j} = \mathbb{E}\left[\left(X_j - \mathbb{E}\left[X_j\right]\right)\left(X_i - \mathbb{E}\left[X_i\right]\right)\right] \tag{A.15}$$

or

$$C_{i,j} = \mathbb{E}\left[X_i X_j\right] - \mathbb{E}\left[X_i\right]\mathbb{E}\left[X_j\right]. \tag{A.16}$$

Written as matrix we get

$$C = \mathbb{E}\left[\left(X - m\right)\left(X - m\right)^T\right]. \tag{A.17}$$

If $C_{i,j} = 0$, the values are called *uncorrelated*.

Auto-correlation
Given A.16, the *auto-correlation* is given by $\mathbb{E}\left[X_i X_j\right]$.

Correlation Coefficient
The *coefficient* is defined over the mean and standard deviation of two random variables as follows

$$corr\left(X, Y\right) = \frac{C}{\delta_X \delta_Y}. \tag{A.18}$$

Mutual Information
The *mutual information* over the PDF of X and Y, given the marginal probabilities is written as

$$I\left(X, Y\right) = \int_X \int_Y p\left(x, y\right) log\left(\frac{p\left(x, y\right)}{p_1\left(x\right) p_2\left(y\right)}\right) dydx. \tag{A.19}$$

Moments
By interpreting a PDF as a mass function, moments can be used as a description. For a density function $p_{x,y}\left(x, y\right)$ *general moments* are given by

$$m_{rs} = \int x^r y^s p_{x,y}\left(x, y\right) dxdy, \text{ for } r, s \geq 0. \tag{A.20}$$

The moments $m_{1,0}$ and $m_{0,1}$ denote the mean values whereas the *central moments*, defined by

$$\mu_{rs} = \int \left(x - \mu_x\right)^r \left(y - \mu_y\right)^s p_{xy}\left(x, y\right) dxdy \tag{A.21}$$

denoting their variance. The mixed *second order moments* are then referring to the covariance

$$\delta_{xy} = \int (x - \mu_x)(y - \mu_y) p_{xy}(x, y) dxdy. \tag{A.22}$$

The *third order moment* describe the skew (asymmetry) around the mean [224] [131].

A.2.2 Rates of Correctness and Precision

True positive rate, also known as recall, hit rate or sensitivity:

$$prob(detectedpositive \mid realpositive) = \frac{truepositive}{truepositive + truenegative}. \tag{A.23}$$

False negative rate also known as miss rate:

$$prob(detectednegative \mid realpositive) = \frac{falsenegative}{truepositive + truenegative}. \tag{A.24}$$

True negative rate also known as specificity or correct rejection rate:

$$prob(detectednegative \mid realnegative) = \frac{truenegative}{truenegative + falsepositive}. \tag{A.25}$$

False positive rate is then:

$$prob(detectedpositive \mid realnegative) = \frac{falsepositive}{truenegative + falsepositive}. \tag{A.26}$$

The precision is

$$prob(detectedpositive \mid realpositive) = \frac{truenegative}{truenegative + falsenegative}. \tag{A.27}$$

The correctness is expressed by

$$prob(truedetected) = \frac{truedetected}{truepositive + falsepositive + truenegative + falsenegative}. \tag{A.28}$$

A widely used combination of measures is the F_1 value:

$$F_1 = 2 * (precision * recall) / (precision + recall). \tag{A.29}$$

[174]

A.2.3 Evaluation Plots

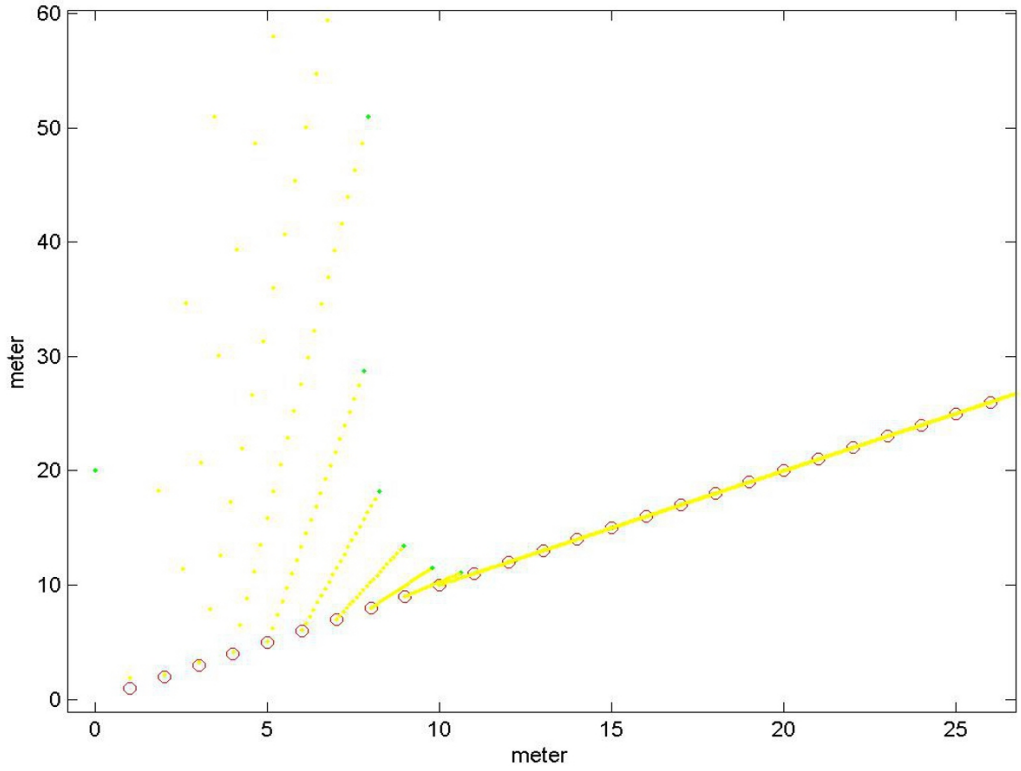

Figure A.9: Illustration of the behavior of the used KF.

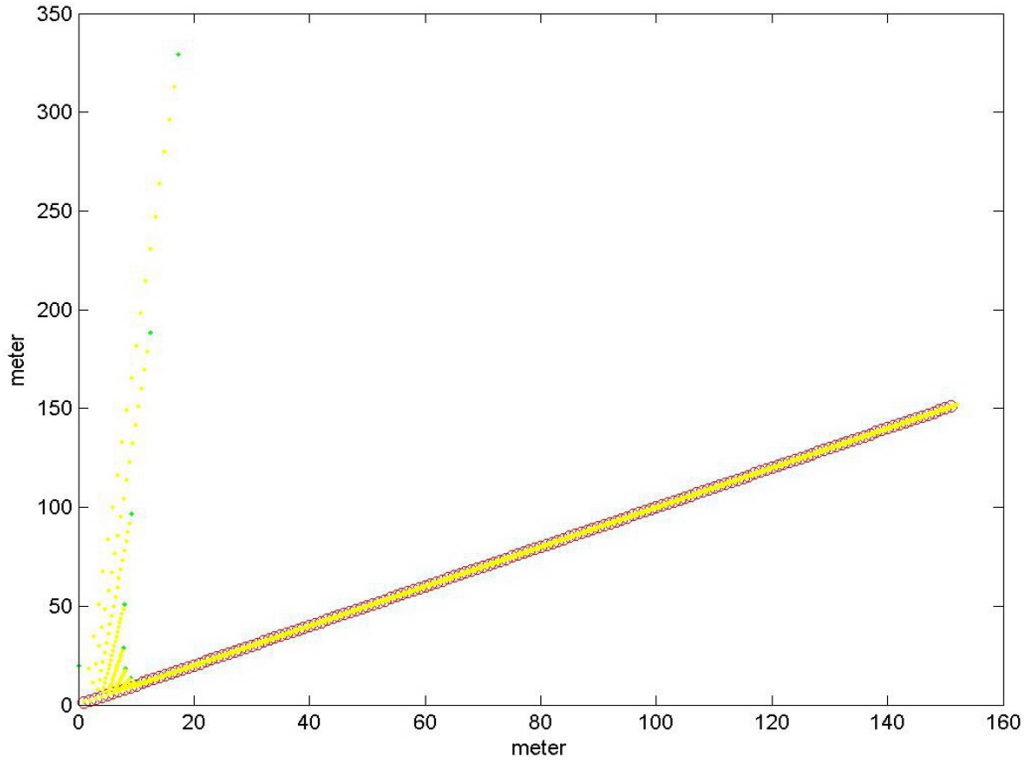

Figure A.10: Illustration of the behavior of the used KF.

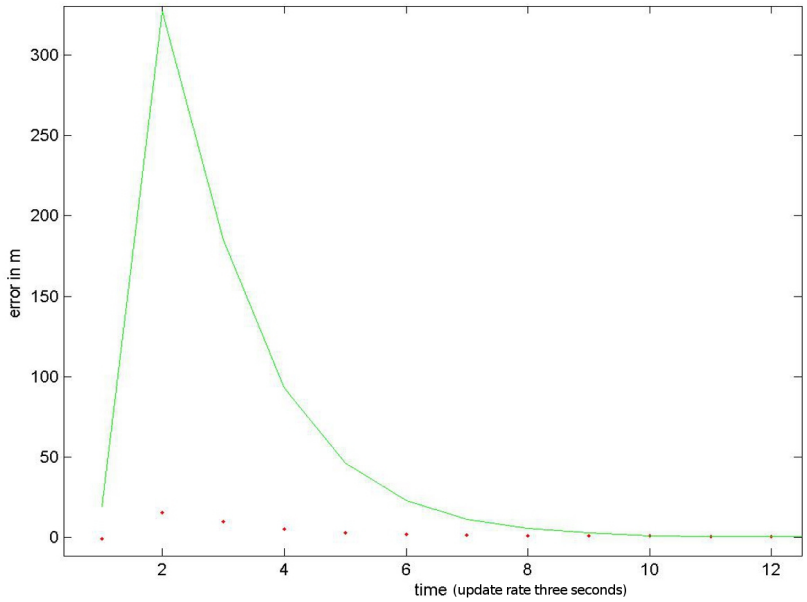

Figure A.11: Illustration of the behavior of the used KF for linear movement.

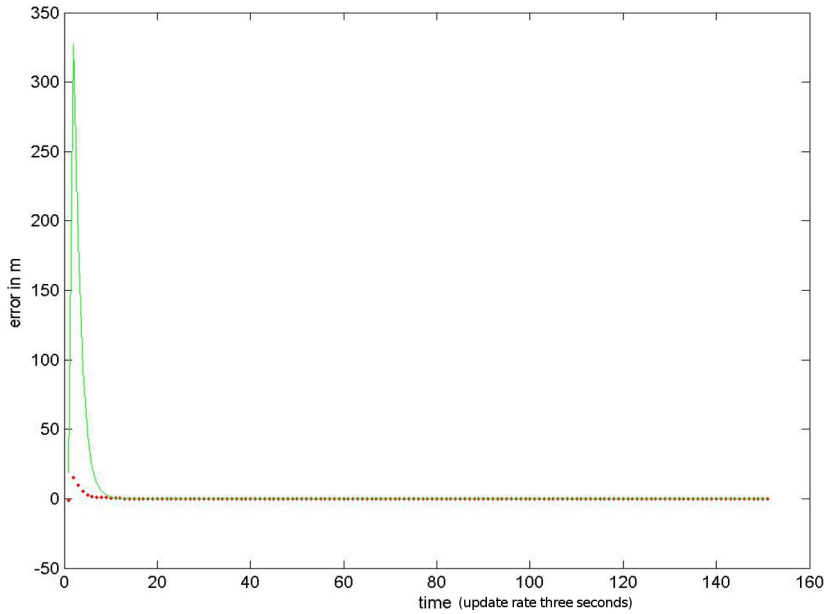

Figure A.12: Illustration of the behavior of the used KF for linear movement.

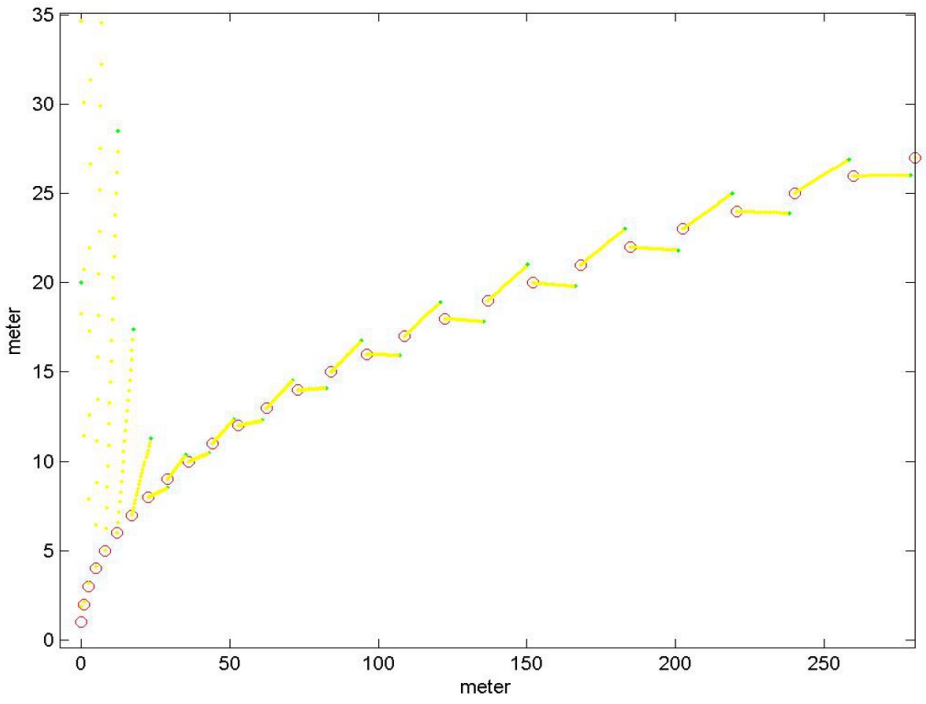

Figure A.13: Illustration of the behavior of the used KF.

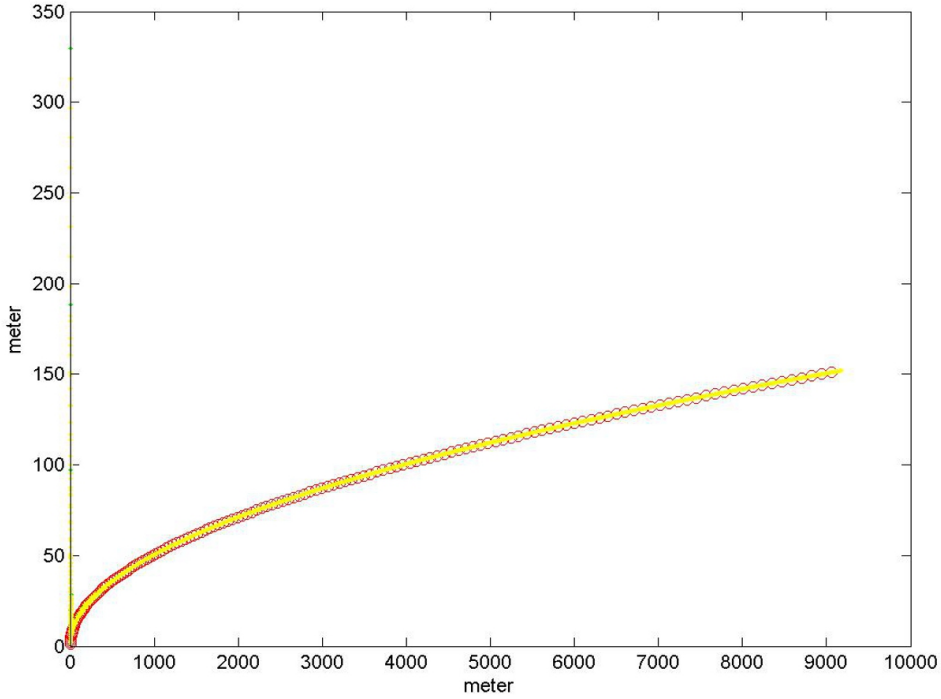

Figure A.14: Illustration of the behavior of the used KF.

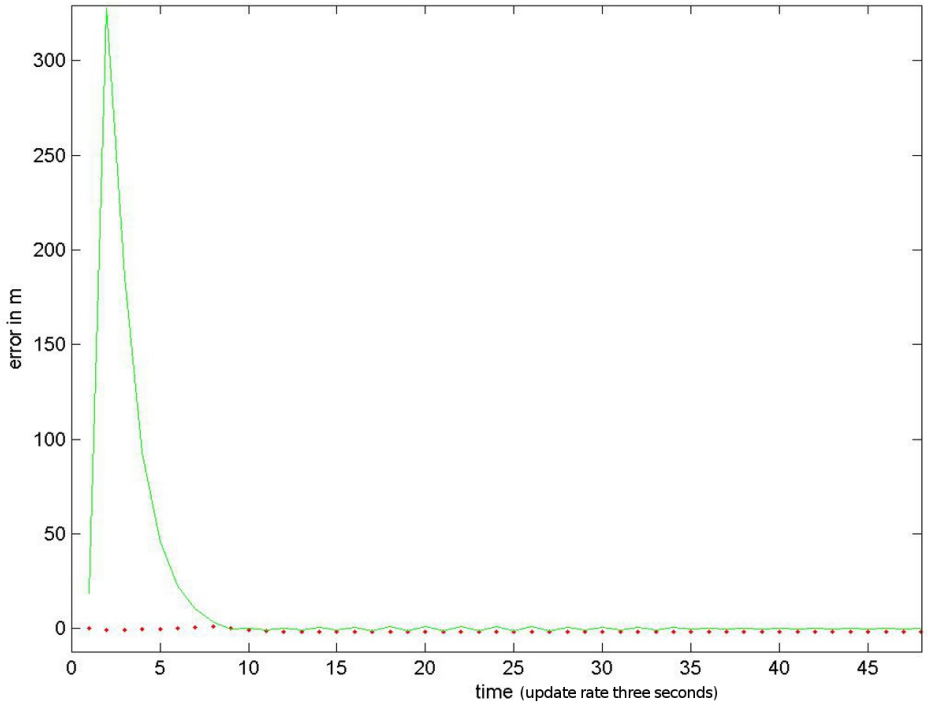

Figure A.15: Illustration of the behavior of the used KF for curved movement.

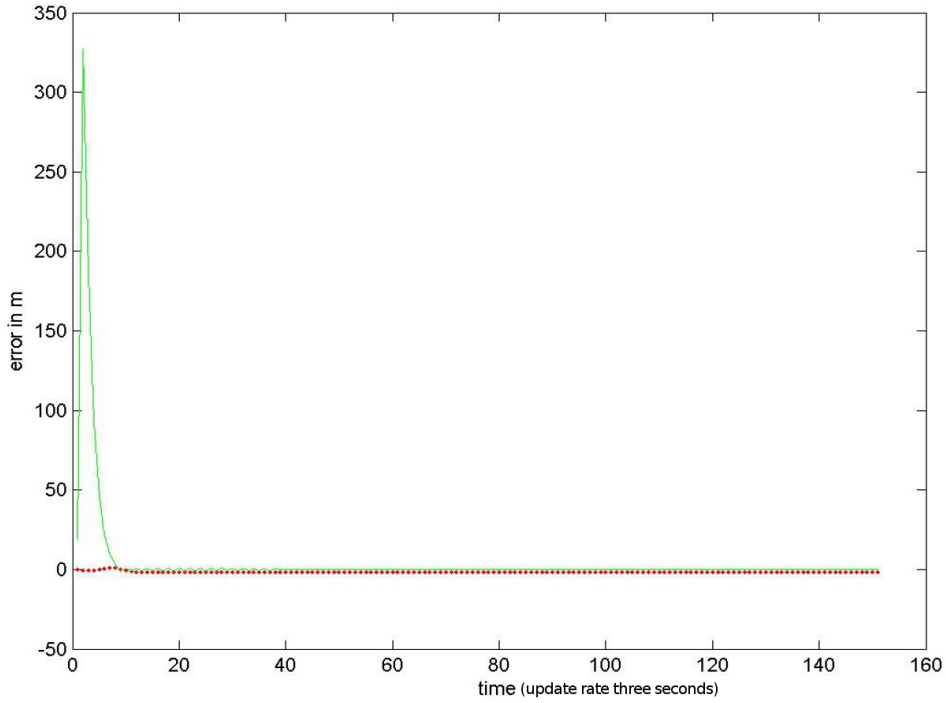

Figure A.16: Illustration of the behavior of the used KF for curved movement.

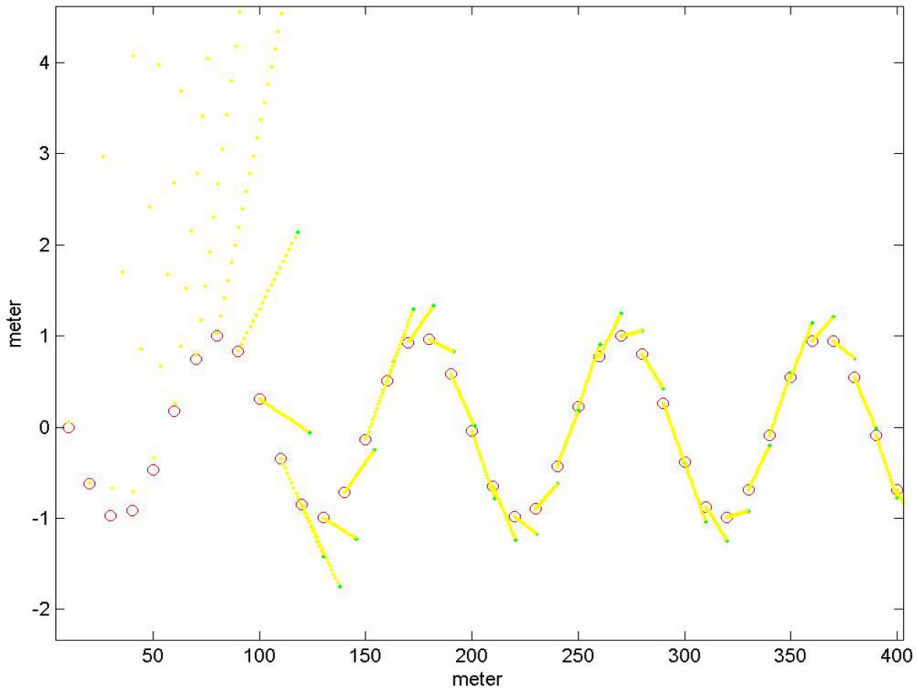

Figure A.17: Illustration of the behavior of the used KF when moving in sinus-like movements.

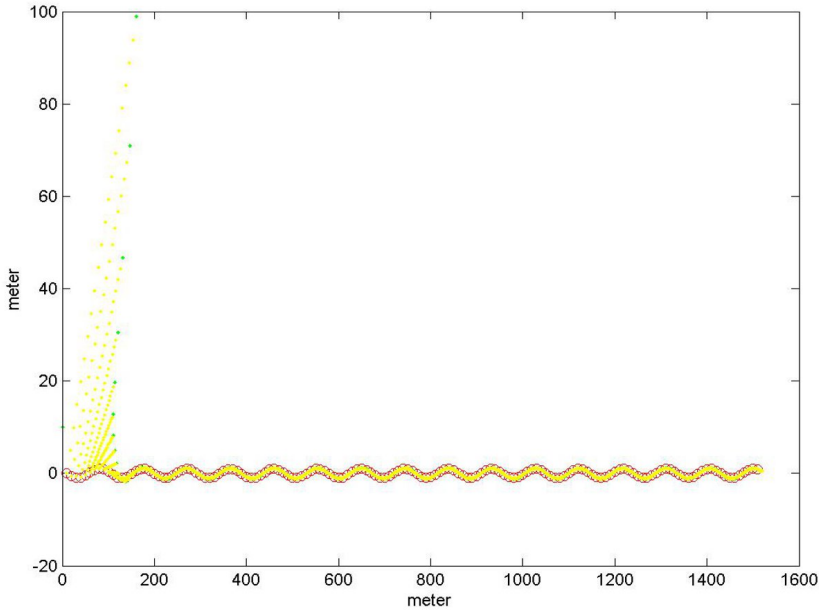

Figure A.18: Illustration of the behavior of the used KF when moving in sinus-like movements.

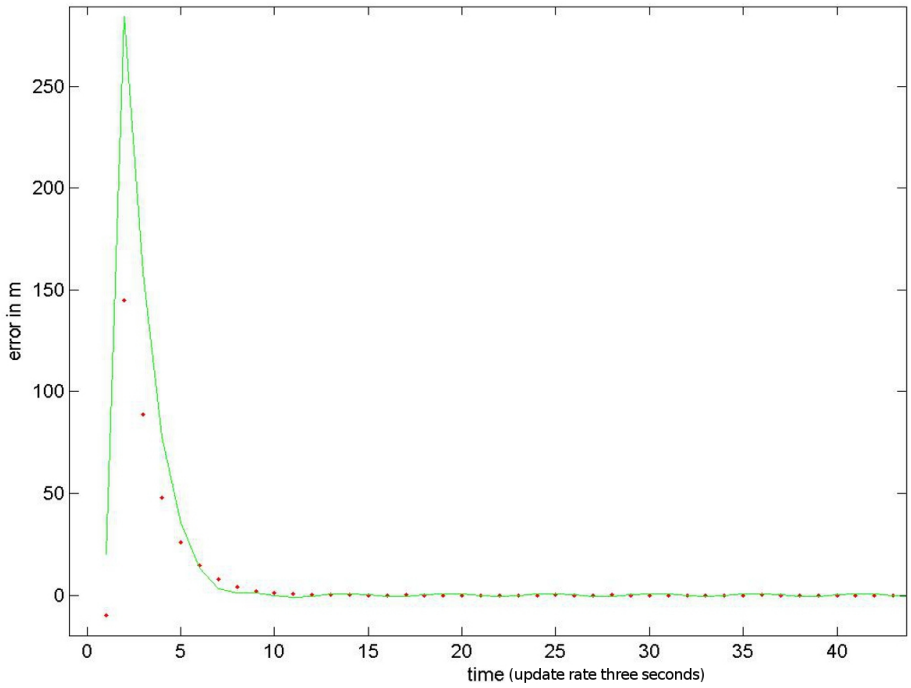

Figure A.19: Illustration of the behavior of the used KF when moving in sinus-like movements.

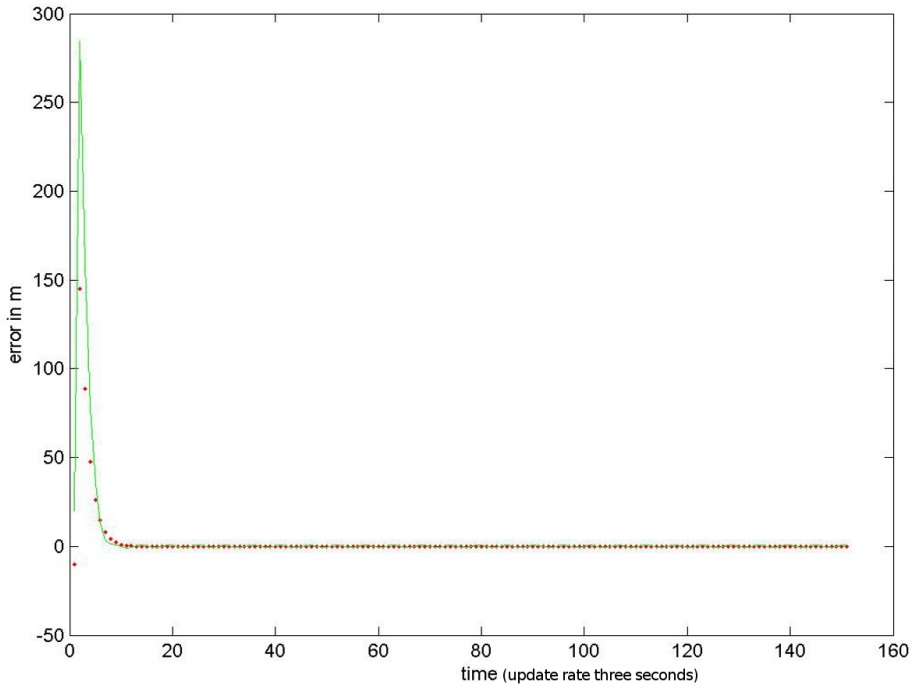

Figure A.20: Illustration of the behavior of the used KF when moving in sinus-like movements.

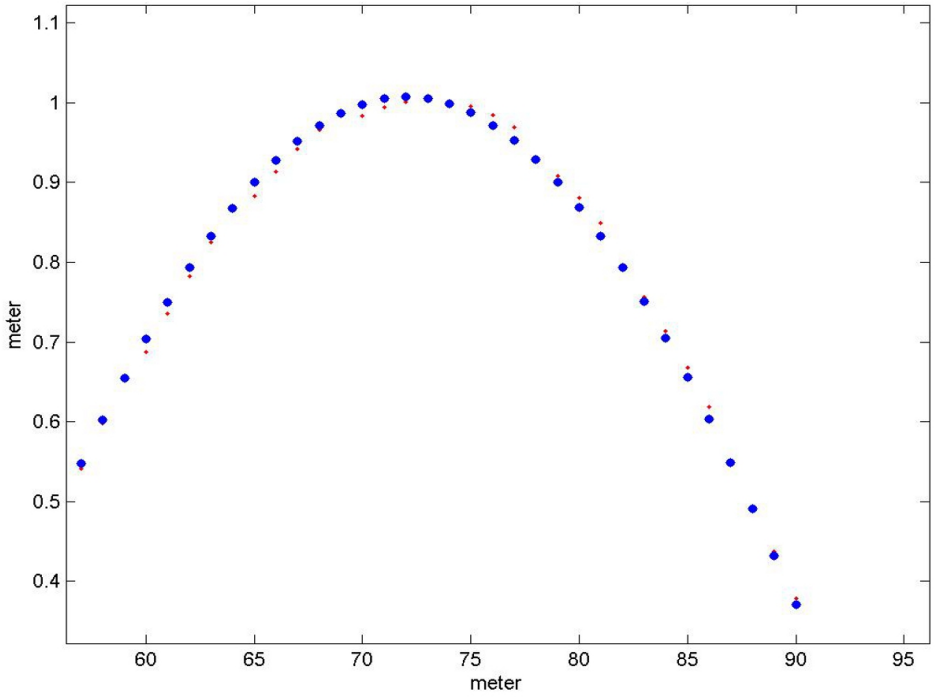

Figure A.21: Results for the prediction of the heading.

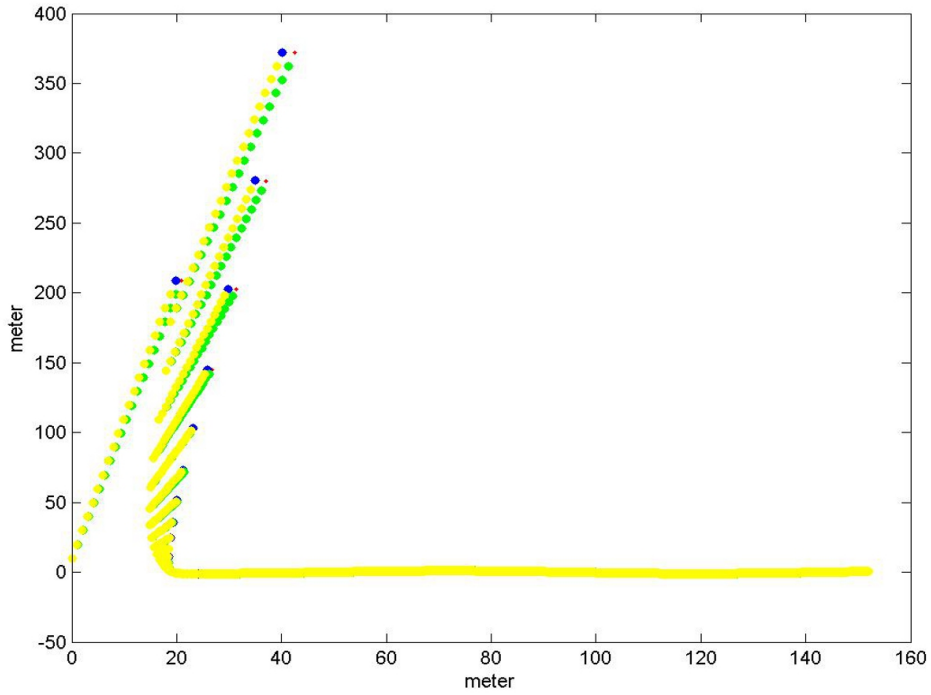

Figure A.22: Results for the prediction of the heading.

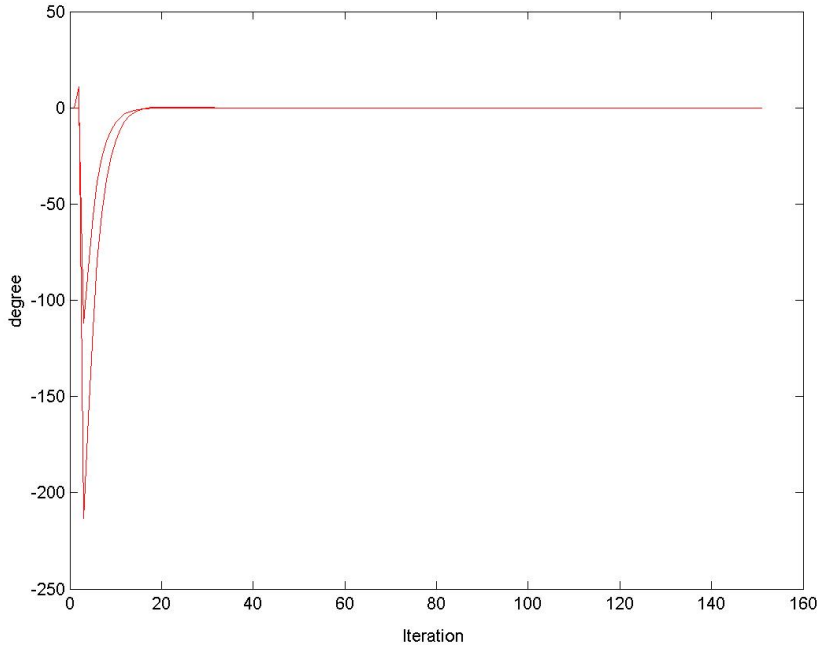

Figure A.23: Results for the prediction of the heading. Plot of the errors in the heading values over time.

A.2.4 List of Common Distributions

The most common distributions used in this statistical, Bayesian context are listed below.

Binomial distribution
This distribution is a function over discrete values. $Bi\left(x \mid \Theta, n\right)$ is defined as

$$p\left(x\right) = \binom{n}{x} \Theta^{x} \left(1-\Theta\right)^{n-x} \tag{A.30}$$

with $0 < \Theta < 1$ for $n = 1, 2, ..., x = 0, 1, 2, ..., n$.

Multinomial distribution
According to A.30, the multinomial distribution is then

$$p\left(x\right) = \frac{n!}{\prod_{i=1}^{k} x_i!} \prod_{i=1}^{k} \Theta_i^{x_i} \tag{A.31}$$

with $0 < \Theta < 1$, for $\sum_{i=1}^{k} \Theta_i = 1, \sum_{i=1}^{k} x_i = n, x_i = 0, 1, .., n$.

Gaussian/Normal distribution
The standard normal distribution is given by

$$p\left(x\right) = \frac{1}{\sqrt{2\pi}} exp\left\{-\frac{x^2}{2}\right\} \tag{A.32}$$

for $-\infty < x < \infty$.

Multivariate normal distribution

$$\mathcal{N}\left(x \mid \mu, \Sigma\right) = \frac{1}{\left(2\pi\right)^{\frac{p}{2}} \left|\Sigma\right|^{\frac{1}{2}}} exp\left\{-\frac{1}{2}\left(x-\mu\right)^{T} \Sigma^{-1} \left(x-\mu\right)\right\} \tag{A.33}$$

with Σ as covariance and μ as mean.

Bibliography

[1] Opengis, city geography markup language encoding. 80

[2] Wikitude gmbh. www.wikitude.com/en/developer/overview/wikitude-sdk. 42

[3] *Merging Real and Virtual Worlds*, IMAGINA, 1995. 36

[4] Marinexml specification document, d11, draft4. http://www.metoc.gov.au/products/xml-schema/marine-xml-schema.html, 2005. Royal Australian Navys Hydrography and METOC Branch,. 83

[5] F. Ababsa, Jean Yves Didier, Malik Mallem, and David Roussel. Head motion prediction in augmented reality systems using monte carlo particle filters. *ICAT*, 2003. 45

[6] Takaaki Endo Akihiro, Akihiro Katayama, Hideyuki Tamura, Michitaka Hirose, Tomohiro Tanikawa, and Makoto Saito. Image-based walk-through system for large-scale scenes. In *4th International Conference on Virtual Systems and MultiMedia - VSMM98*, 1998. 44

[7] Lee Alexander, Michael Brown, and Barry Greenslade. The next edition if iho s-57 (edition 4): Much more than encs. *ICC, International Conference on Communications*, 2005. 83

[8] K. Ali, D. Fleuret, F. Hasler, and P. Fua. Joint pose estimator and feature learning for object detection. *IEEE International Conference on Computer Vision*, 2009. 45

[9] Android. Android sdk. http://developer.android.com/sdk/index.html. 42

[10] M. Austin. Creating a gis from noaa electronic navigational charts. In *OCEANS, Proceedings of MTS/IEEE*, volume 1, pages 839–841. NOAA Coast Survey Development Laboratory, 2005. ISBN: 0-933957-34-3. 46

[11] R. Azuma, B. Hoff, H. Neely, and R. Sarfaty. A motion-stabilized outdoor augmented reality system. *Proceedings of IEEE Virtual Reality*, pages 252–259, 1999. 43

[12] Ronald Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments 6*, 4:355–385, 1997. ix, 36, 39, 41, 108

[13] Ronald T. Azuma. A survey of augmented reality. *Course Notes 9: Developing Advanced Virtual Reality Applications, ACM SIGGRAPH*, pages 1–38, 1995. 35

[14] Ronald T. Azuma. The challenge of making augmented reality work outdoors. *Mixed Reality: Merging Real and Virtual Worlds.*, pages 379–390, 1999. pages 379–390, Springer. 42

[15] Neil M. Barbour. Inertial navigation sensors. *Sensors & Electronics Technology (SET), Research and TechnologyOrganisation NATO*, 2010. RTO-EN-SET-116(2010). 14

[16] J. Barrett. Side effects of virtual environments: A review of the literature. Technical Report DSTO-TR-1419, DSTO Information Sciences Laboratory PO Box 1500 Edinburgh South Australia 5111 Australia, 2004. 40

[17] Jeffrey O. Bauer. Modern developments in geometry. Wayne State College, Modern Developments in Geometry, 2005. 10

[18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008. 45

[19] Reinhold Behringer. Registration for outdoor augmented reality applications using computervision techniques and hybrid sensors. *VR '99 Proceedings of the IEEE Virtual Reality*, 1999. ISBN:0-7695-0093-5. 46

[20] Christoph Beierle and Gabriele Kern-Isberner. *Methoden wissensbasierter Systeme*. Computational Intelligence. Vieweg + Teubner, 4. edition, 2008. 26

[21] G. Biegelbauer and M. Vincze. Efficient 3d object detection by fitting superquadrics to range image data for robots object manipulation. *IEEE International Conference on Robotics and Automation*, 2007. 46

[22] Oliver Bimber and Ramesh Raskar. *Spatial Augmented Reality Merging Real and Virtual Worlds*. A K Peters, Ltd. Wellesley, Massachusetts, 2005. ISBN 1-56881-230-2. ix, 36, 37, 38

[23] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer Verlag, 2006. ISBN-10: 0-387-31073-8. ix, 19, 20, 25, 28, 30, 33, 98

[24] Ian Bishop and P. Ghadirian. Integration of augmented reality and gis: A new approach to realistic landscape visualization. *Landscape and Urban Planning*, 86:226–232, 2008. 46

[25] Gabriele Bleser and Gustaf Hendeby. Using optical flow as lightweight slam alternative. *IEEE International Symposium on Mixed and Augmented Reality, Science and Technology Proceedings*, 2009. 43

[26] Casey Boardman. Pearl' s belief propagation algorithm and loopy bayesian networks. *Networks*, 2004. 27

[27] Alexander Borghgraef et al. An evaluation of pixel-based methods for the detection of floating objects on the sea surface. *EURASIP Journal on Advances in Signal Processing*, 2010. Article ID 978451. 42

[28] I. N. Bronstein, K. A. Semendjajew, G. Grosche, and E. Zeidler. *Teubner-Taschenbuch der Mathematik*. Number 1 in Teubner-Taschenbuch der Mathematik. Teubner, 2003. 136

[29] F. P. Brooks. The computer scientist as toolsmith ii. *CACM 39*, 3:61–68, 1996. 36

[30] S. Brown et al. Head-up display for automobile, 04 1988. 41

[31] Berking Jonas Buettgenbach, Hecht. *Die elektronische Seekarte*. Wichmann Verlag, 1999. isbn: 3-87907-303-1. ix, 57, 58, 91

[32] Ben Butchart. Augmented reality for smartphones. Technical report, University of Bristol, JISC, 03 2011. Full draft. 42

[33] C.J. Butz, J. Chen, K. Konkel, and P. Lingras. A comparative study of variable elimination and arc reversal in bayesian network inference. *Twenty Second International Florida Artificial Intelligence Research Society Conference (FLAIRS)*, pages 571–572, 2009. 28

[34] S. Calderara, A. Prati, and R. Cucchiara. Hecol: Homography and epipolar-based consistent labeling for outdoor park surveillance. *Computer Vision and Image Understanding, Elsevier*, 2008. 45

[35] Robert O. Castle and David W. Murray. Object recognition and localization while tracking and mapping. *IEEE International Symposium on Mixed and Augmented Reality Science and Technology Proceedings*, 2009. 45

[36] T.P. Caudell and D.W. Mizell. Augmented reality: an application of heads-up display technology to manual manufacturing processes. *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences*, ii:659, 1992. Print ISBN: 0-8186-2420-5, Boeing Comput. Services, Seattle. 37

[37] B. Cha, H. Kawano, N. Suetake, and T. Aso. Fast minimum-spannung-tree-like image segmentation. *IEEE Fourth International Conference on Natural Computation*, 2008. 99

[38] Lin Chai, William A. Hoff, William A. Hoff corresponding, and Tyrone Vincent. 3-d motion and structure estimation using inertial sensors and computer vision for augmented reality. *Presence*, 11:474–492, 2000. 43

[39] C. H. Chen. *Handbook Of Pattern Recognition And Computer Vision*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2005. 10, 29, 34

[40] Kar Wee Chia, Adrian David Cheok, and Simon J.D. Prince. Online 6 dof augmented reality registration from natural features. *International Symposium on Mixed and Augmented Reality (ISMAR02)*, 2002. 44

[41] Andrew I. Comport, Eric Marchand, and Francois Chaumette. A real-time tracker for markerless augmented reality. *ACM/IEEE Int. Symp. on Mixed and Augmented Reality*, pages 36–45, 2003. 46

[42] R. Cowell, G. Dawid, A. Philip Lauritzen, L. Steffen, and David J. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Statistics for Engineering and Information Science. Springer, 1999. ISBN 0-387-98767-3. 15, 16, 20, 26, 28

[43] Fabio Cozman, Eric Krotkov, and Carlos Guestrin. Outdoor visual position estimation for planetary rovers. *Auton. Robots*, 9:135–150, September 2000. 46

[44] Boguslaw Cyganek. A real-time vision system for traffic signs recognition invariant to translation, rotation and scale. *ACIVS, LNCS*, 5259:278–289, 2008. 99

[45] Paul Dagum and Michael Luby. Approximating probabilistic inference in bayesian belief networks is np-hard. *Artif. Intell.*, 60:141–153, March 1993. 26

[46] Adnan Darwiche. A differential approach to inference in bayesian networks. In *Journal of the ACM*, pages 123–132, 2000. 28

[47] Andrew J. Davison et al. Monoslam: Real-time single camera slam. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6), 2007. 43

[48] E. Deistung, A. Friedewald, B. Mesing, and C. Schaefer. Szenarien fuer den einsatz von vr-technologien in der maritimen wirtschaft. *3D-NordOst 2007: 10. Anwendungsbezogener Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D Daten*, pages S. 123–130, 2007. Berlin: Gesellschaft zur Foerderung angewandter Informatik e.V. 42

[49] Deeptendu Bikash Dhar and Bhabatosh Chanda. Extraction and recognition of geographical features from paper maps. *International Journal of Document Analysis*, 8:232–245, 2006. 46

[50] Thomas G. Dietterich. Machine learning for sequential data: A review. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 15–30. Springer-Verlag, 2002. 105

[51] Prof. Dillmann and M. Loesch. Einordnungskriterien fuer ml-verfahren. Universitaet Karlsruhe, Vorlesung Maschinelles Lernen, January 2009. 17

[52] Clark Dodsworth. Smartphone augmented reality context is king. IEEE CNSV, April 20 2010. 42

[53] Ichiro; Liao Hongen Dohi, Takeyoshi; Sakuma, editor. *Medical Imaging and Augmented Reality, 4th International Workshop Tokyo*. Lecture Notes in Computer Science, Vol. 5128. Springer, image processing, computer vision, pattern recognition, and graphics edition, 2008. ISBN 978-3-540-79981-8. 41

[54] Marek J. Druzdzel and Henri J. Suermondt. Relevance in probabilistic models: "backyards" in a "small world". In *Working notes of the AAAI-1994 Fall Symposium Series: Relevance*, pages 60–63, 1994. 29

[55] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, 11 2001. 23

[56] S.R. Ellis. What are virtual environments. *Computer Graphics and Applications, IEEE*, 14 Issue: 1:17–22, 1994. 36

[57] Caarls et al. Sensor fusion for augmented reality. *EUSAI*, LNCS 2875:160–176, 2003. 44

[58] Kerstin Falkowski, Juergen Ebert, Peter Decker, Stefan Wirtz, and Dietrich Paulus. Semi-automatic generation of full citygml models from images. *Geoinformatik*, Bd. 35, 2009. 87, 126

[59] J. Feltens. Vector methods to compute azimute, elevation, ellipsoidal normal, and thecartesian (x,y,z) to geodetic transformation. *J. of Geod., Springer Verlag*, 2007. 54

[60] Univ.-Prof. Mag. Dr. Alois Ferscha. Embedded and pervasive systems, sensor technologies. Institute for Pervasive Computing, Johannes Kepler University. Vorlesung SS2011. 38

[61] Graham Finlayson and Gerald Schaefer. Hue that is invariant to brightness and gamma. In *Proc. British Machine Vision Conference*, pages 303–312, 2001. 100

[62] J.T. Fischer. *Rendering Methods forAugmented Reality.* PhD thesis, Eberhard-Karls-Universitaet Tuebingen, 2006. 40

[63] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach.* Prentice Hall, 2003. ISBN-10: 0130851981. 10, 98, 99, 133

[64] Y. Furukawa and J. Ponce. Accurate camera calibration from multi-view stereo and bundle adjustment. *Int J Comput Vis*, 84:257–268, 2009. 39

[65] Donald S. Fussell. Advanced image synthesis. The University of Texas at Austin, Nov 2007. CS 395T Fall 2007. 10

[66] R. Girshick. Object detection with heuristic coarse-to-fine search. *Illinois Computer Vision Workshop, TTI-C*, 2009. 46

[67] O. Grau, A. Thomas, A. Hilton, and J. Stark. A robust free-viewpoint system video for sport scenes. *International Conference CAIP, Computer Analysis of Images and Patterns*, 1995. 42

[68] O. Grau, R. Toenjes, and R. Koch. Analyse durch synthese modellierung von 3d objekten in stereobildfolgen. *1. Workshop visual computing, Darmstadt*, March 1994. 46

[69] TU Graz. Studierstube augmented reality for collaborative and ubiquitous computing. 42

[70] J.G. Grimes. Global position system standard positioning service performance standard. Department of Defense, USA, September 2008. 4th Edition. 10

[71] Haipeng Guo and William Hsu. A survey of algorithms for real-time bayesian network inference. In *the joint AAAI-02/KDD-02/UAI-02 workshop on Real-Time Decision Support and Diagnosis Systems*, 2002. 26, 28

[72] K. Haase and R. Koch. Augmented reality system for nautical navigation. *Lecture Notes in Informatics Series, GI*, Mobile and Embedded Interactive Systems, 2008. ISBN 978-3-88579-227-7. 5

[73] K. Haase and R. Koch. Augmented reality fernglas zur navigationsunterstützung, 2009. 5

[74] K. Haase and R. Koch. Extension of electronical nautical charts for 3d interactive visualization via citygml. *Proceedings of GeoInformatik, Heidelberg*, 2010. ISBN 978-3-89838-335-6. 5

[75] K. Haase and R. Koch. Extension of sea charts for 3d visualization. *ISPRS proceedings*, 3DGeoInfo, 2010. 5

[76] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, ISBN: 0521623049, 2000. ix, 8, 9, 10, 136

[77] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition.* Springer Series in Statistics. Springer, 2nd edition, 9 2009. 31

[78] Morton L. Heilig. Sensorama simulator, August 1962. United States Patent Office, PatNr. 3050870. 37

[79] E.E. Hemayed. A survey of camera self-calibration. *IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS*, 2003. 39

[80] Axel Hildebrand. Augmented reality in medical applications. Fraunhofer Inst. fuer Graphische Datenverarbeitung (IGD); Korea Univ.; Korea Advanced Inst. of Science and Technology; Samsung Biomedical Research Inst.: Advanced Medical Image Processing 1998, Seoul; Proceedings. Session III, No. 3. Hrsg.: M.-H. Kim, H.-, Januar 1998. 41

[81] Prof. H. Hillmer and M. Bartels. Grundwissen der elektronik 10 - vergleich cmos ccd. Skriptum, Universitaet Kassel, 01 2008. 133

[82] S. Hinterstoisser, S. Benhimane, and N. Navab. N3m: Natural 3d markers for real-time object detection and pose estimation. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–7, 2007. 45

[83] Thuong N. Hoang and Bruce H. Thomas. In-situ refinement techniques for outdoor geo-referenced models using mobile ar. *IEEE International Symposium on Mixed and Augmented Reality*, 2009. 45

[84] Jeroen Hol. *Pose Estimation and Calibration Algorithms for Vision and Inertial Sensors.* PhD thesis, Department of Electrical Engineering, Linkoeping University, 2008. 60

[85] Christopher D. Hollander and Annie S. Wu. The current state of normative agent-based systems. *Journal of Artificial Societies and Social Simulation*, 14(2):6, 2011. 26

[86] Richard L. Holloway. Registration error analysis for augmented reality. Presence Teleoperators and Virtual Environments, Volume: 6, Issue: 4 1997. ISSN: 10547460. 108

[87] Dezhong Hong, Jian-Kang Wu, and Sumeet Sohan Singh. Refining image retrieval based on context-driven methods. In *International Conference: IS&T/SPIE's 11th International Symposium on Electronic Imaging '99*, pages 581–593, 1999. 45

[88] HP. Hp webos palm. https://developer.palm.com/. 42

[89] Hsin-Cheng Huang. Introduction to statistical image analysis using markov random fields. Institute of Statistical Science, 10 2010. http://www.math.ntu.edu.tw/ hchen/jointseminar/. 22

[90] Mike Hueftle. Stochastische prozesse in der zeitreihenanalyse. Lehrstuhl fuer Wirtschaftsinformatik und Operations Research Rheinisch-WestfaelischenTechnischen Hochschule Aachen, 7 2006. OptiV Project. 21

[91] IALA. Iala guidelines on the universal automatic identification system (ais) volume 1, part ii technical issues edition 1.1, 2002. 12

[92] IALA. Iala technical clarifications on itu recommendation itu-r m.1371-1 edition 1.4. Association of Internationale de Signalisation Maritime, International Association of Marine Aids to Navigation and Lighthouse Authorities, 2003. 12, 13

[93] IALA. Iala guideline no. 1028, on the automatic identification (ais) volume 1, part i operational issues edition 1.3, 2004. 13

[94] Seung-bin Im and Sung-bae Cho. Context-based scene recognition using bayesian networks with scale-invariant feature transform. *Networks*, 4179:1080–1087, 2006. 125

[95] Apple Inc. mobile operating system sdk. https://developer.apple.com/ipad/sdk/, July 2011. 42

[96] N. Iwasaki, Y. Kawaguchi and Y. Inagaki. Design, implementation and evaluations of a direction based service system forboth indoor and outdoor. *UCS 2004*, LNCS 3598:20–36, 2005. 43

[97] Bernd Jaehne. *Digitale Bildverarbeitung.* Springer Berlin, 2005. ISBN-13: 9783540249993 ISBN-10: 3540249990. 133

[98] F. V. Jensen. Bayesian network basics. *A I S B Quarterly*, No. 94:9–22, 1996. 26

[99] Finn Jensen. *An Introduction to Bayesian Networks.* London, UCL Press, 1996. ISBN-10: 0387915028. 25, 26, 98

[100] S. Julier, Y. Baillot, M. Lanzagorta, D. Brown, and L. Rosenblum. Bars: Battlefield augmented reality system. In *NATO Symposium on Information Processing Techniques for Military Systems*, pages 9–11, 2000. 41

[101] S. Julier et al. Information filtering for mobile augmented reality. *IEEE Computer Graphics and Applications*, 22:12–15, 2002. 40

[102] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME Journal of Basic Engineering*, (82 (Series D)):35–45, 1960. 53

[103] H. Kaufmann. *Geometry Education with Augmented Reality*. PhD thesis, TU Wien, 2004. 42

[104] Hannes Kaufmann and Dieter Schmalstieg. Mathematics and geometry education with collaborative augmented reality. *Computers and Graphics*, 27:339–345, 2003. 42

[105] J. Keuchel, M. Heiler, and C. Schnoerr. Hierarchical image segmentation based on semdefinite programming. *DAGM, LNCS 3175*, pages 120–128, 2004. 99

[106] A. D. KING. Inertial navigation - forty years of evolution. *GEC REVIEW*, 13(3):140–149, 1998. ix, 13, 14

[107] Gary R. King, Wayne Piekarski, and Bruce H. Thomas. Arvino outdoor augmented reality visualisation of viticulture gis data. In *Proceedings of the forth IEEE and ACM International conference on Mixed and Augmented Reality (ISMAR), oct 5-8, Wienna*, pages 52–55, 2005. 41, 43

[108] R. Koch et al. Architecture and tracking algorithms for a distributed mobile industrial ar system. *The 5th ICVS07*, Bielefeld, 2007. 41, 45

[109] Thomas H. Kolbe et al. Making interoperability persistent: A 3d geo database based on citygml. *Lecture Notes in Geoinformation and Cartography: 3D Geo Information Science*, 2009. 83

[110] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. Adaptive computation and machine learning. MIT Press, 2009. 15, 18, 19, 25, 98

[111] F. Krumm and V. S. Schwarze. *Schriftenreihe der Institute des Studiengangs Geodaesie und Geoinformatik, Technical reports*. Universitaet Stuttgart, 1999. ISSN 0933-2839 Report Nr. 1999.6. 21

[112] F. Kurugollu, B. Sankur, and A.E. Harmanci. Color image segmentation using histogram multithresholding and fusion. *Image and Vision Computing*, 19:915–928, 2001. 100

[113] Wearable Computer Lab. Tinmith project. School of Computer and Information Science, University of South Australia. 42

[114] P. F. Lammertsma. Satellite navigation. Technical report, Institute of Information and Computing Sciences Utrecht University, Feb 2005. ix, 11

[115] Richard O. Lane et al. Maritime anomaly detection and threat assessment. *13th International Conference on Information Fusion*, 2010. 42

[116] V. Lepetit et al. Fast keypoint recognition using random ferns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:448–461, 2010. 45

[117] O. Lezoray. Color image segmentation using morphological clustering and fusionwith automatic scale selection. *Pattern Recogn. Lett.*, 30:397–406, 2009. 99

[118] Olivier Lezoray and Hubert Cardot. Bayesian marker extraction for color watershed in segmenting microscopicimages. *Pattern Recognition, Proceedings. 16th International Conference on*, 1:739–742, 2002. 99

[119] Yan Lin and Marek J. Druzdzel. Computational advantages of relevance reasoning in bayesian belief networks. In *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence*, pages 342–350. Morgan Kaufmann, 1997. 29

[120] Enno Littmann and Helge Ritter. Adaptive color segmentation- a comparision of neural and statistical methods. *IEEE Transactions on neural networks*, 8(1):175–185, January 1997. 99

[121] M. A. Livingston and Zhuming Ai. The effect of registration error on tracking distant augmented objects. In *Mixed and Augmented Reality, 2008. ISMAR 2008. 7th IEEE/ACM International Symposium on*, pages 77–86, 2008. 3

[122] Mark A. Livingston, Lawrence J. Rosenblum, Simon J. Julier, Dennis Brown, Yohan Baillot, J. Edward, Swan Ii, Joseph L. Gabbard, and Deborah Hix. An augmented reality system for military operations in urban terrain. In *In Interservice/Industry Training, Simulation, and Education Conference*, page 89, 2002. 41

[123] L.D. Lopez and O. Fuentes. Color-based road sign detection and tracking. *ICIAR, LNCS 4633*, pages 1138–1147, 2007. 99

[124] David G. Lowe. Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*, 2:1150–1157, 1999. 45

[125] Chien-Ping Lu and Eric Hager, Gregory D. andMjolsness. Fast and globally convergent pose estimation from video images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:610–622, 2000. 46

[126] Blair Macintyre, Enylton Machado Coelho, and Simon J. Julier. Estimating and adapting to registration errors in augmented reality systems. In *in IEEE Virtual Reality Conference 2002 (VR 2002)*, pages 73–80. IEEE Press, 2002. 108

[127] J. P. Marques de Sa. *Pattern Recognition, Concepts, Methods and Applications*. Springer, 2001. ISBN-10: 3540422978. 17, 33

[128] J. Matas, O. Chum, M. Urba, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *Proc. of British Machine Vision Conference*, pages 384–396, 2002. 45

[129] Bruce A. Maxwell and S.A. Shafer. Physic-based segmentation of complex objects using multiple hypothesis of image formation. *Computer Vision and Image Understanding*, 65(2):269–295, 1997. 99

[130] Peter S. Maybeck. *Mathematics in science and engineering: Stochastic models, estimation adn control*. Academic Press Inc., 1979. isbn 0-12-480701-1. 52

[131] J. Chris McGlone. *Manual of Photogrammetry, Fifth Edition*. asprs- American Society for Photogrammetry and Remote Sensing, 2004. isbn 1-57083-071-1. 136, 138

[132] S. Meers, K. Ward, and I. Piper. Simple, robust and accurate head-pose tracking using a single camera. *The Thirteenth Annual Conference on Mechatronics and Machine Visionin Practice*, December 2006. Toowoomba, Australia. 42

[133] D. Meintrup and S. Schaeffler. *Stochastik, Theorie und Anwendungen*. Springer, 2005. ISBN-10: 3540216766. 21

[134] James E. Melzer, Kaiser Electro-Optics Inc. *The Avionics Handbook*, chapter 5 - Head-Mounted Displays. CRC Press LLC, 2001. 41

[135] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(27):1615–1630, 2005. 45

[136] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65:2005, 2005. 99

[137] P. Milgram and F. Kishino. A taxonomy of mixed reality visual displays. *IEICE Transactions on Information Systems*, Vol E77-D(No.12), December 1994. 36

[138] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino. Augmented reality: A class of displays on the reality-virtuality continuum. In *SPIE Vol. 2351, Telemanipulator and Telepresence Technologies*, pages 282–292, 1994. ix, 36

[139] Daniel D. Morris, Brian R. Colonna, and Franklin D. Snyder. Image-based motion stabilization for maritime surveillance. *proceedings of SPIE - Volume 6497, Image Processing: Algorithms and Systems V*, 2007. Eds. Jaakko T. Astola, Karen O. Egiazarian, Edward R. Dougherty. 46

[140] J. Neyman and E. S. Pearson. *Joint Statistical Papers*. Hodder Arnold, 1967. University of California Press, Berkeley, ISBN 0852647069. 26

[141] Y-Y. Nguwi and Z. K. Abbas. Detection and classification of road signs in natural environements. *Neural Comput & Applic*, 17:265–289, 2008. 100

[142] Andreas Nüchter and Joachim Hertzberg. Towards semantic maps for mobile robots. *Robot. Auton. Syst.*, 56:915–926, November 2008. 45

[143] N. Oltersdorf, M. Schindler, and G. Thavanesan. Technologische grundlagen und anwendungsfelder von augmented reality. Fachhochschule für Oekonomie und Management in Duesseldorf, 2010. 41

[144] International Hydrographic Organization. Specifications for chart content and display aspects of ecdis. International Hydrographic Bureau MONACO. Special Publication No. 52. 57

[145] D. Parikh and C. Tsuhan. Unsupervised learning of hierarchical semantics of objects. In *Computer Vision and Pattern Recognition*, pages 1–8, june 2007. 45

[146] D. Paulus. Structure from motion. Universitaet Koblenz, May 2003. Exercises and Tutorials by S. Bouattour and D. Droege. 10, 135

[147] D. Paulus et al. Danger sign detection using color histograms and surf matching. *Safety, Security and Rescue Robotics, SSRR, IEEE International Workshop on*, pages 13–18, 2008. 100

[148] Windows Phone. Windows mobile 7. www.microsoft.com/windowsphone/de. 42

[149] Justus H. Piater and Roderic A. Grupen. Feature learning for recognition with bayesian networks. In *In Proceedings of the Fifteenth International Conference on Pattern Recognition. IEEE*, 2000. 125

[150] W. Piekarski. *Interactive 3D Model ling in Outdoor Augmented Reality Worlds.* PhD thesis, University of South Australia, 2004. 39

[151] Wayne Piekarski and Bruce Thomas. Arquake: the outdoor augmented reality gaming system. *Magazine Communications of the ACM - Internet abuse in the workplace and Game engines in scientific research*, 45(Issue 1), 2002. 41

[152] J. Pilet, C. Strecha, and P. Fua. Making background subtraction robust to sudden illumination changes. *ECCV, LNCS*, 5305:567–580, 2008. 99

[153] Playstation. Playstation developer kit. http://de.playstation.com/ps3/. 42

[154] M. Plumlee, C. Ware, R. Arsenault, and R. Brennan. Panoramic images for situational awareness in a 3d chart-of-the-future display. *U.S. Hydro*, 2005. 42

[155] Thomas Porathe. If this will be the way to drive a ship - just anyone could do it: 3d nautical charts. about creating acceptance and building standards for a vr within the maritime domain. *SIGRAD*, 2010. 2

[156] Thomas Porathe. 3d computer graphics and nautical charts. *Go-3D Conference*, 978-3-8396-0275-1, 2011. 2

[157] Federico Prandi, Raffaella Brumana, and Francesco Fassi. Semi-automatic objects recognition in urban areas based on fuzzy logic. *Journal of Geographic Information System*, 2:55–62, 2010. 45

[158] L. Prasad and A.N. Skourikhine. Vectorized image segmentation via trixel agglomeration. *GbRPR, Springer*, 3434:12–22, 2005. 99

[159] M. Pressigout and E. Marchand. Model-free augmented reality by virtual visual servoing. *IAPR Int. Conf. on Pattern Recognition*, 2004. 46

[160] J. Prison and T. Porathe. Navigation with 2d- and 3d- maps: A comparative study with maritime personnel. *39th Nordic Ergonomy Society Conference*, 2007. 2

[161] W. Qi, F. Li, and L. Zhenzhong. Review on camera calibration. *IEEE Control and Decision Conference, China*, 2010. 39

[162] L-R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77, 1989. 23

[163] A. Radburn. A mobile augmented reality demonstrator. *Proceedings, First International Workshop on Mobile Geospatial Augmented Reality*, 2006. 41, 42

[164] Y. Raja, S. McKenna, and S. Gong. Segmentation and tracking using colour mixture models. In *In Asian Conference on Computer Vision*, pages 607–614, 1998. 99

[165] A. Rankin and L. Matthies. Daytime water detection based on color variation. *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2010. 100

[166] Clarence E. Rash. A 25-year retrospective review of visual complaints and illusions associated with a monocular helmet-mounted display. *Displays, Elsevier B.V.*, 29:70–80, 2008). 41

[167] Clarence E. Rash, Michael B. Russo, Tomasz R. Letowski, and Elmar T. Schmeisser. *Helmet-Mounted Displays: Sensation, Perception and Cognition Issues*. U.S. Army Aeromedical Research Laboratory Fort Rucker, Alabama, 2009. ISBN 978-0-615-28375-3. 41

[168] R Raskar, G. Welch, and H. Fuchs. Spatially augmented reality. *Proc. IWAR98, IEEE*, San Francisco, CA, USA, 1998. 40

[169] Holger Regenbrecht, Gregory Baratoff, and Wilhelm Wilke. Augmented reality projects in the automotive and aerospace industries. *IEEE Comput. Graph. Appl.*, 25:48–56, November 2005. 41

[170] G. Reitmayr and T.W. Drummond. Going out: robust model-based tracking for outdoor augmented reality. In *Mixed and Augmented Reality, 2006. ISMAR 2006. IEEE/ACM International Symposium on*, pages 109–118, oct. 2006. 45

[171] G. Reitmayr and D. Schmalstieg. Collaborative augmented reality for outdoor navigation and. In *In Proceedings of the Symposium on Location Based Services and TeleCartography*, pages 31–41. Wiley, 2004. 46

[172] G. Reitmayr and D. Schmalstieg et al. Pose tracking from natural features on mobile phones. *IEEE International Symposium on Mixed and Augmented Reality*, 2008. 45

[173] G. Reitmayr and D. Schmalstieg et al. Real-time detection and tracking for augmented reality on mobile phones. *IEEE transaction on visualization and computer graphics*, 16, 2010. 45

[174] C.J.V. Rijsbergen. *Information retrieval.* Butterworths, 1979. ISBN 0-408-70929-4. 138

[175] G. W. Roberts, A. Evans, and A. H. Dodson. The use of augmented reality, gps and ins for subsurface data visualization. *FIG XXII Internat. Congress*, 2002. 42, 43

[176] C. Rother et al. Image segmentation by branch-and-mincut. *ECCV*, IV:15–29, 2008. 99

[177] O. Rusch, C. Ruwwe, and U. Zoelzer. Image segmentation in naval ship images. *11. Workshop Farbbildverarbeitung*, 2005. 99

[178] H. Samet and A. Soffer. A legend-driven geographic symbol recognition system. In *Pattern Recognition, 1994. Vol. 2 - Conference B: Computer Vision Image Processing., Proceedings of the 12th IAPR International.*, volume 2, pages 350–355, Oct 1994. 46

[179] K. Satoh, K. Hara, M. Anabuki, H. Yamamoto, and H. Tamura. Townwear: An outdoor wearable mr system with high-precision registration. *Proc. Virtual Reality 2001. IEEE, Yokohama, Japan*, pages 210–211, 2001. 41

[180] F. Scalzo and J. Piater. Statistical learning of visual feature hierarchies. *IEEE Conference on Computer Vision and Pattern Recognition CVPR*, 2005. 99

[181] Gerhard Schall et al. Handheld augmented reality for underground infrastructure visualization. *Pers Ubiquit Comput*, 13:281–291, 2009. 43

[182] Milo Schield and Thomas V. V. Burnham. Von mises frequentist approach to probability. Section on Statistical Education JSM 2008. 26

[183] D. Schmalstieg and H. Bischof et al. Multiple target detection and tracking with guaranteed framerateson mobile phones. *IEEE International Symposium on Mixed and Augmented Reality*, 2009. 45

[184] D. Schmalstieg and G. Reitmayr. Augmented reality as a medium for cartography. *Multimedia Cartography*, 2nd ed. (ed. W. Cartwright, M. Peterson, G. Gartner):267–282, 2006. 38, 40, 46

[185] Dieter Schmalstieg and Gerhard Reitmayr. The world as a user interface: Augmented reality for ubiquitous computing. 2007. 40

[186] Dieter Schmalstieg and Gerhard Reitmayr et al. Global pose estimation using multi-sensor fusion for outdoor augmentedreality. *IEEE International Symposium on Mixed and Augmented Reality*, 2009. 44

[187] Dieter Schmalstieg, Gerhard Schall, Daniel Wagner, Istvn Barakonyi, G. Reitmayr, Joseph Newman, and Florian Ledermann. Managing complex augmented reality models. *IEEE Computer Graphics and Applications*, 272:32–41, 2007. 40

[188] D. Schmalstieg et al. Wide area localization on mobile phones. *IEEE International Symposium on Mixed and Augmented Reality*, 2009. 45

[189] H. Schneiderman. Learning a restricted bayesian network for object detection. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR.*, volume 2, pages 639–646, 2004. 125

[190] Paul Schnitzspan, Mario Fritz, and Bernt Schiele, editors. *Hierarchical Support Vector Random Fields Joint Training to Combine Local and Global Features*, Marseille, France, 10 2008. 125

[191] C. Schulte and V. Coors. Development of a citygml ade for dynamic 3d flood information. *Joint ISCRAM-CHINA and GI4DM Conference on Information Systems on Information Systems for Crisis Management, Harbin, China*, 2008. 83

[192] B. Schwerdtfeger and G. Klinker et al. Pick-by-vision: A first stress test. *IEEE International Symposium on Mixed and Augmented Reality 2009 Science and Technology Proceedings*, 2009. 41

[193] S. Shah and J. K. Aggarwal. Mobile robot navigation and scene modelling using stereo fish-eye lens system. *Machine Vision and Applications*, 10 Issue 4, 1997. 46

[194] J. Shotton et al. Real-time human pose recognition in parts from single depth images. *to appear*. Microsoft Research Cambridge & Xbox Incubation. 41

[195] Aarti Singh and Larry Wasserman. Statistical machine learning. Lectures course in advanced machine learning, Carnegie Mellon, spring 2011. 23

[196] U. Steinhoff, B. Schiele, and A. Leonardis et al. How computer vision can help in outdoor positioning. *AmI07 Proceedings of the 2007 European conference on Ambient intelligence, Springer*, pages 124–141, 2007. 44

[197] J.P. Stenbit. Global position system standard positioning service performance standard. Department of Defense, USA, Oktober 2001. 4th Edition. 11

[198] Didier Stricker. Tracking with reference images: A real-time and markerless tracking solution for out-door augmented reality applications. *Proceedings of the conference on Virtual reality, archeology, and cultural heritage*, 2001. 44

[199] D. Stricker et al. The matris project: Real-time markerless camera tracking for augmented reality. *Journal of Real-time Imageprocessing*, 2(2-3):69–79, 2007. 41

[200] T. Sumimoto and K Kuramoto. Detection of a particular object from motion images under bad conditions. *IEEE Instrumentation and Measurement Technology Conference*, 1:318–322, 2001. 100

[201] Ian E. Sutherland. A head-mounted three dimensional display. *Proceeding AFIPS '68 (Fall, part I)*, December 9-11 1968. ACM New York, NY, USA. 36, 37

[202] G. Sziebig. Achieving total immersion: Technology trends behind augmented reality - a survey. *Proceedings of the 9th WSEAS International Conference on Simulation, Modelling and Optimization*, pages 458–463, 2009. ISSN: 1790-2769. 38, 39

[203] T. Tamati et al. Knowledge -based picture understanding of weather charts. *Pattern Recognition*, 17:109–123, 1984. 46

[204] Li Tan. *Digital Signal Processing: Fundamentals and Applications.* 1 Edition. Academic Press, 2007. ISBN 0-12-374090-8. 133

[205] Tang, Kwoh, Teo, Sing, and Ling. Augmented reality systems for medical applications. *Engineering in Medicine and Biology Magazine, IEEE*, 17(3):49–58, 2002. 41

[206] C.P. Town and D. Sinclair. Content based image retrieval using semantic visual categories. Technical report, 2001. 45

[207] Volker Tresp. Bayesche netze: Konstruktion, inferenz, lernen und kausalitaet. *Ludwig-Maximilians-Universitaet Muenchen, Department of Computer ScienceDatabase Systems Group, Lecture in Machine Learning and Data Mining*, 2011. 17, 25

[208] Alexander A. Trusov. Overview of mems gyroscopes: History, principles of operations, types of measurements. *MicroSystems Laboratory, Mechanical and Aerospace Engineering University of California, Irvine, CA, 92697, USA*, May 2011. 14

[209] Fuan et al. Tsai. Texture generation and mapping using video sequences for 3d building models. *Lecture Notes in Geoinformation and Cartography: Innovations in 3D Geo Information Systems*, 2006. 87, 126

[210] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. *Computer Engineering*, 3952:589–600, 2006. 99

[211] H. Uchiyama et al. Ar gis on a physical map based onmap image retrieval using llah tracking. *IAPR Conference on Machine Vision Applications*, 12(3), 2009. 46

[212] Human Interface Technology Laboratory (HIT Lab) University of Washington. Artoolkit. 42

[213] S. S. Vaez, A. Rajabifard, A. Binns, and I. P. Williamson. Developing a seamless sdi model across the land-sea interface. *Coast GIS07*, 2007. 46

[214] Christoffer Valgren and Achim J. Lilienthal. Sift, surf & seasons: Appearance-based long-term localization in outdoor environments. *Robotics and Autonomous Systems*, 58:149–156, 2010. 45

[215] A. van den Hengel, A. Dick, T. Thormaehlen, B. Ward, and P. H. S. Torr. A shape hierarchy for 3d modelling from video. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, GRAPHITE 07, pages 63–70, New York, NY, USA, 2007. ACM. 46

[216] Hendrik van Hees. Statistische physik. Institut fuer Theoretische Physik Justus-Liebig-Universitaet Giessen, 11 2008. http://theory.gsi.de/ vanhees/faq-pdf/stat.pdf. 21

[217] J. Venn. *The Logic of Chance*. Kessinger Publishing, 2006. 26

[218] Jaume Vergas Llaha. *Color Constancy and Image Segmentation Techniques for Applications to Mobile Robotics*. PhD thesis, Universitat Politacnica de Catalunya. Departament d'Enginyeria de Sistemes, Automatica i Informatica Industrial, 2005. 99

[219] V. Vlahakis, N. Ioannidis, J. John Karigiannis, M. Tsotros, and M. Gounaris. Virtual reality and information technology for archaeological site promotion. In *Proc. 5th International Conference on Business Information Systems (BIS02*, pages 24–25, 2002. 41

[220] John von Neumann. Various techniques used in connection with random digits. monte carlo methods, 12 1951. 44

[221] Daniel Wagner. History of mobile augmented reality. Christian Doppler Laboratory for Handheld Augmented Reality. ISMAR society, https://www.icg.tugraz.at/ daniel/HistoryOfMobileAR/. 37

[222] B. Walsh. Introduction to bayesian analysis. *Lecture Notes for EEB 596z*, 2002. 17, 25

[223] Y. Wang, D. Wang, and W. Fang. Color segmentation and part model matching for non-rigid objects tracking. *International Conference on Computer Science and Software Engineering*, pages 911–914, 2008. 99

[224] Andrew Webb. *Statistical Pattern Recognition.* JohnWiley & Sons, Ltd, 2004. ISBN 0-470-84513-9. 16, 17, 30, 34, 138

[225] M. Weis, S. Mueller, C.-E. Liedtke, and M. Pahl. A framework for gis and imagery data fusion in supportof cartographic updating. *Information Fusion*, 6:311–317, 2005. 46

[226] Y. Weiss and J. Pearl. Belief propagation and perspectives. Technical Report R-368, University of California, Los Angeles, July 2010. Forthcoming, Communications of the ACM. 27

[227] Greg Welch and Gary Bishop. An introduction to the kalman filter. *SIGGRAPH 2001*, Update 2006. 52

[228] Jan Wendel. *Integrierte Navigationssysteme, Sensordatenfusion, GPS und Intertiale Navigation.* Oldenbourg Verlag Muenchen Wien, 2007. isbn: 978-3-486-58160-7. 10, 14, 24, 52

[229] Stefan Wuerz and Karl Rohr. *Pattern Recognition, Lecture Notes in Computer Science*, volume 3175/2004, chapter 3D Segmentation and Quantification of Human Vessels Based on a New 3D Parametric Intensity Model, pages 111–119. Springer Berlin / Heidelberg, 2004. 56

[230] Jieping Ye. Least squares linear discriminant analysis. In *Proceedings of the 24th international conference on Machine learning*, ICML '07, pages 1087–1093, New York, NY, USA, 2007. ACM. 31

[231] S. You, U. Neumann, and R. Azuma. Hybrid inertial and vision tracking for augmented reality registration. pages 260–267, 1999. 44

[232] Suya You and Ulrich Neumann. Fusion of vision and gyro tracking for robust augmented reality registration. *IEEE Virtual Reality Conference*, 2001. 44

[233] T.Y. Young and K.-S. Fu. *Handbook of Pattern Recognition and Image Processing.* Academic Press Inc., 1986. isbn: 0-12-774560-2. 133

[234] David Yuen and Bruce A. Macdonald. Natural landmark based localization system using panoramicimages. *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, 1:915–920, 2002. 44

[235] Dong-Qing Zhang. *Statistical Part-Based Models: Theory and Applications in Image Similarity, Object Detection and Region Labeling.* PhD thesis, 2005. 125

[236] Zhengyou Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–198, 1998. 9

[237] Wei Zhang et al. Video compass. *ECCV 2002*, LNCS 2353:476–490. 45

[238] ZKR. Inland ais geraete auf binnenschiffen nach dem standard schiffsverfolgung und aufspaerung in der binnenschifffahrt, 31.5. 2007. 13

[239] Jean-Marie Zogg. Gps locate, communicate, accelerate essentials of satellite navigation compendium. u-blox AG, 2009. GPS X 02007 D, Theory and Principles of Satellite Navigation. ix, 10, 12