Doctoral Thesis

Location estimation in indoor environments using time-of-flight range camera

Author(s):
Kohoutek, Tobias K.

Publication Date:
2012

Permanent Link:
https://doi.org/10.3929/ethz-a-007618337

Rights / License:
In Copyright - Non-Commercial Use Permitted
LOCATION ESTIMATION IN INDOOR ENVIRONMENTS USING A TIME-OF-FLIGHT RANGE CAMERA

A dissertation submitted to

ETH ZURICH

for the Degree of
Doctor of Sciences

presented by

TOBIAS KARL KOHOUTEK

Dipl.-Ing., TU Dresden

born February 11, 1981 in Frankfurt (Oder), Germany

citizen of Germany

accepted on the recommendation of

Prof. Dr.-Ing. Hilmar Ingensand, ETH Zurich
Dr. Jan Böhm, University College London (UCL)
Prof. Bertrand Merminod, EPF Lausanne

2012
ABSTRACT

Indoor positioning with different technical solutions is omnipresent in industrial and academic research. The most important applications are Location Based Services (LBS), which objects require reference in a coordinate system. Research and development target for example the automation of processes in smart warehousing and logistics, or the monitoring of people during rescue missions. Indoor positioning is also highly relevant to robotics and autonomous navigation. The poor performance of Global Navigation Satellite Systems (GNSS) in indoor environments calls for other solutions. Diverse requirements and different environmental conditions, in particular Non-Line-of-Sight (NLoS) signal propagation, are reasons for the current insufficient level of performance in indoor positioning and navigation. Wireless devices (e.g. RFID systems) enjoy widespread use in numerous diverse applications including sensor networks, deployed in all environments and organizing themselves in an ad-hoc fashion. However, knowing the correct positions of network nodes and their deployment is an essential precondition. Optical sensors do not require the deployment of physical reference infrastructure inside buildings and offer several solutions covering all required accuracy levels.

The aim of this thesis is to apply range images from a Time-of-Flight (ToF) range camera for indoor positioning. Range Imaging (RIM) is a special technique in the spectra of electro-optic and video-metric principles. It is capable to capture the environment three-dimensionally in real-time. Single camera systems offer a high potential for indoor applications. Camera position and possible movements can be derived after insignificant details have been eliminated. Furthermore, semantic information can be extracted from the purely metrical data using geometric constraints to establish a connection between the spatio-semantic information of installations and objects in the scene.

This thesis is based on five scientific publications, which have been framed, by an introduction and a concluding chapter. Publication 1 focuses on the localization and tracking/monitoring of a robot. Publication 2 describes human computer interaction based motion detection of people. Publications 3 to 5 concentrate on the location estimation of a ToF range camera itself in a scene compared to a spatio-semantic interior building model. Such models can be referenced to any arbitrary coordinate system. The proposed approach can therefore be used for absolute positioning of objects/installations and human operators in real time with centimeter accuracy. However, the camera position in relation to surrounding objects, which are compared with their database models, is derived with decimeter accuracy. Simultaneous Localization And Mapping (SLAM) generates 3D modeled environments in the proposed method.
ZUSAMMENFASSUNG


# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
</tr>
<tr>
<td>ZUSAMMENFASSUNG</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
</tr>
<tr>
<td>LIST OF FIGURES AND TABLES</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
</tr>
<tr>
<td>Shortcomings and Research Questions</td>
</tr>
<tr>
<td>Objectives</td>
</tr>
<tr>
<td>Overview of the Thesis</td>
</tr>
<tr>
<td>2 BACKGROUND</td>
</tr>
<tr>
<td>Robot Monitoring for Online Collision Avoidance</td>
</tr>
<tr>
<td>Tracking of Body Parts for Human Computer Interaction</td>
</tr>
<tr>
<td>Camera Based Location Estimation in Indoor Environments</td>
</tr>
<tr>
<td>3 OBJECT TRACKING USING A TOF RANGE CAMERA</td>
</tr>
<tr>
<td>Publication 1 - Analysis and processing of 3D-Range-Image-data for robot monitoring</td>
</tr>
<tr>
<td>Publication 2 - Multi-user vision interface based on Range Imaging</td>
</tr>
<tr>
<td>4 INDOOR POSITIONING USING A TOF RANGE CAMERA</td>
</tr>
<tr>
<td>Publication 3 - Real-time indoor positioning using Range Imaging sensors</td>
</tr>
<tr>
<td>Publication 4 - Fusion of Building Information and Range Imaging for Autonomous Location Estimation in Indoor Environments</td>
</tr>
<tr>
<td>Publication 5 - Indoor positioning and navigation using Time-of-Flight cameras</td>
</tr>
<tr>
<td>5 CONCLUDING REMARKS</td>
</tr>
<tr>
<td>Synopsis of results</td>
</tr>
<tr>
<td>Outlook</td>
</tr>
<tr>
<td>ACKNOWLEDGMENT</td>
</tr>
<tr>
<td>ABBREVIATIONS</td>
</tr>
<tr>
<td>REFERENCES</td>
</tr>
<tr>
<td>CURRICULUM VITAE</td>
</tr>
</tbody>
</table>
LIST OF FIGURES AND TABLES

Figures

2 BACKGROUND

Fig. 1 Multi-robot system. 5
Fig. 2 Fish tank VR system. 6
Fig. 3 Yamabico M. 8
Fig. 4 Semantic 3D Building Information Model. 10

3 OBJECT TRACKING USING TOF RANGE CAMERA

Publication 1

Fig. 1 Robot working space. 11
Fig. 2 Phase delay between the emitted and the reflected signal: a) phase delay; b) modulated signal. 12
Fig. 3 Experimental setup. 13
Fig. 4 Edge detection algorithms. 14
Fig. 5 Robot edges and features. 14
Fig. 6 Template for edge features. 14
Fig. 7 Security zones; a) static, b) dynamic. 15
Fig. 8 Monitored areas. 15
Fig. 9 Motion field vs. Optical Flow; a) Motion of the sphere without grey value changing; b) Motion of the light source with grey value changing. 16
Fig. 10 Displacement vector field at Optical Flow. 16
Fig. 11 a) Aperture problem, b) Image pyramid. 17
Fig. 12 Detected object. 17

Publication 2

Fig. 1 Phase delay between the emitted and the reflected signal: a) phase delay; b) modulated signal. 21
Fig. 2 View frustum and virtual touchscreen. 21
Fig. 3 Hand-head line model. 22
Fig. 4 3D scene (mirror-inverted) and top view to the same scene. 24
Fig. 5 Face detection in distance of 3 m; faces towards the camera (left), one face with small rotation (right). 24
4 INDOOR POSITIONING USING A TOF RANGE CAMERA

Publication 3

Fig. 1 ETH Zurich lecture room HIL C71.3 in CityGML. 29
Fig. 2 Today’s positioning systems in dependence to accuracy and coverage. 30
Fig. 3 ToF cameras, left: PMD Technologies: CamCube 2.0, right: MESA® Imaging: SR4000. 32
Fig. 4 Object comparison between a range image (left) and form primitives from database (right). 33

Publication 4

Fig. 1 Overall concept of the presented approach. 38
Fig. 2 Acquired point cloud from ToF range camera. 40
Fig. 3 Origin (x, y, z) as delivered by the camera. 41
Fig. 4 VRML model of the test object with seven surfaces and ten object points (left), as wire frame model (middle) and with added random points (right). 42
Fig. 5 Time-of-flight principle. 43
Fig. 6 Final test results after calibration by manufacturer. 43
Fig. 7 Acquired and model point cloud after \( T^o_r \) translation (SR4000 point cloud without floor in white and model point cloud in red). 45
Fig. 8 Solution of NDT algorithm (input data: object model point cloud in green and SR4000 point cloud in white). 45
Fig. 9 Solution of CG algorithm from different viewing angles: filtered SR4000 point cloud (white) and GIS object model point cloud (green). 46
Fig. 10 Multiple path reflections on a concave scene - in that case a corner between two walls. 46

Publication 5

Fig. 1 Today’s positioning systems in dependence to accuracy and coverage. 50
Fig. 2 Decision tree for room identification. 52
Fig. 3 ETH Zurich lecture room modeled in CityGML. 52
Fig. 4 Phase unwrapping of an indoor scene. (a) Image of the scene. (b) and (c) 3D point clouds that have been generated based on the camera’s depth image. Color of the points indicates the result of the algorithm; wrapped measurements are shown in red. Bright Brightness encodes distance to the camera center. (b) Point cloud without unwrapping. Measured distances beyond the sensor’s non-ambiguity range are wrapped into it, which results in artifacts between distances of 0 and 3 meters. (c) Unwrapped depth image. 54
Fig. 5  SIFT feature extraction and matching applied on two consecutive camera frames on a ToF reflectance image. The numbers of detected features are 475 (a) and 458 (b). (c) Matching result: 245 features from image (a) are matched to features from image (b). White lines indicate feature displacement.

Fig. 6  The resulting 3D map based on the estimated trajectory (red). The colors of the points correspond to the distance of the point from the ground plane.

Fig. 7  (a) 3D Point cloud of an exemplary scene. The color of the points corresponds to the distance, brighter color relates to shorter distances and darker color to farther distances. (b) The generated height image. The grayscale value of every pixel corresponds to the z-coordinate of the respective point in the point cloud. (c) The resulting obstacle points (red).

Fig. 8  The resulting virtual scan of the scene is compared with the scan from the laser range finder. The dashed green line illustrates the base laser scan. The red line illustrates the virtual laser scan. The chair shows only a few points in the base laser scan since only the legs of the chair are in the scan plane, whereas the virtual scan outlines the contour of the chair.
Tables

1 INTRODUCTION

Table 1 Overview of state-of-the-art RIM sensors with active illumination. 2

2 BACKGROUND

Table 2 Classification criteria for 3D models of indoor space. 9

4 INDOOR POSITIONING USING A TOF RANGE CAMERA

Publication 4

Table 1 SR4000 specifications. 43
Table 2 Parameters of NDT used to produce Fig. 8. 45
Table 3 Code example for the used CG algorithm. 45
1 INTRODUCTION

Humans and most animals have the capability to acquire spatial information of a scene by stereo vision (Richter, 2012). Cognitive perception begins with “global evaluation” in which the overall impression of the scene is acquired. Subsequently, a detail analysis is carried out where the information is categorized and the attention is focused on striking objects or relevant parts of the scene. Insignificant details are filtered out and will not be further processed. Practice and experience optimize the selection and sequence of the important details (Helmholtz, 1867). It is a natural process for humans to locate their position in indoor environments relative to doors, windows and any kind of objects.

The absolute location estimation of an object, its monitoring as well as the monitoring of its surroundings became increasingly important for many applications in all environments. Global Navigation Satellite Systems (GNSS) cover the positioning requirements, like overall availability and centimeter accuracy for outdoor applications. However, these systems perform poorly indoors. Due to the importance to cover the indoor environment for automatization or monitoring, various alternative approaches have been developed. Mautz (2012) presents a survey through state-of-the-art indoor positioning methods. Indoor positioning systems covering ranges from 1 – 2000 m with typical accuracies between micrometers to meters are detailed and categorized into 13 technologies. For many applications the required positioning accuracy is millimeters to centimeters. Geodetic methods, such as total stations or rotational lasers (e.g. iGPS (Müller et al., 2006)), reach this level of accuracy. Vision based methods have become a dominating technique in indoor positioning and navigation. They can be categorized in static sensors or ego-motion systems (Mautz and Tilch, 2011). The image coordinates reflect angular measurements that can be used to determine the 3D pose of a device by spatial resection. In a further step the position can be referenced to a 3D building model or floor plan, like it is presented in Kitanov et al. (2007) or Hile and Borriello (2008).

This thesis is based on range finding techniques (Jarvis, 1983) as an optical indoor positioning system. State-of-the-art sensors with active illumination can be categorized in Time-of-Flight (ToF) and Triangulation devices. Time-Correlated Single Photon Counting (TCSPC) in Single Photon Avalanche Diodes (SPAD) or phase shift measurements in Photonic Mixer Devices (PMD) are used for image acquisition in ToF sensors. Triangulation technique became a consumer application for completely hands-free control of electronic devices with the launch of Microsoft’s Kinect™ in 2010. Kinect’s depth sensor consists of an infrared laser projector combined with a monochrome Complementary Metal–Oxide–Semiconductor (CMOS) sensor. The 3D coordinate of the observed target is calculated in a stereovision system by triangulation. Popescu et al. (2003) presented a hand-held prototype for scene modeling using a video camera and attached 16 laser pointers. The system worked with five frames per second registering the color and triangulation data. Three years later Popescu et al. (2006) presented an advanced version with a pattern of 7 x 7 laser beams and a progressive scan mode of 15 frames per second resulting in a depth error of 3.5 cm of a target in 3 m distance. However, in contrast to observing the object with a ToF range camera, the object must be visible for projector and camera at the same time and the length of the stereo baseline directly drives the performance of a stereo system. These three techniques vary vastly in their properties, which is presented in a brief overview in Table 1.
Table 1 Overview of state-of-the-art RIM sensors with active illumination

<table>
<thead>
<tr>
<th>Properties</th>
<th>Single photon counting</th>
<th>Phase shift measurement</th>
<th>Triangulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor size (pixel)</td>
<td>32 x 32</td>
<td>max. 204 x 204</td>
<td>640 x 480</td>
</tr>
<tr>
<td>Frame rate (frames/second)</td>
<td>49000</td>
<td>up to 90</td>
<td>30</td>
</tr>
<tr>
<td>Range (m)</td>
<td>3</td>
<td>(0.8) up to 20</td>
<td>0.8 – 3.5</td>
</tr>
<tr>
<td>Depth resolution (mm)</td>
<td>&lt; 2</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(for ranges &lt; 10m)</td>
<td>(at 2 m range)</td>
</tr>
<tr>
<td>Example</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Due to their high acquisition rate so called Range Imaging (RIM) sensors (Besl, 1988) collect three-dimensional (3D) data of a real-world scenes. Marszalec et al. (1995) used a Light Emitting Diode (LED) array based sensor for 3D profile measurement. R. Lange and Seitz (2001) presented a solid-state ToF range camera that enables to measure 3D Cartesian coordinates up to a distance of 10 m. Ranges are measured for each individual pixel using only one camera. The actual coordinate accuracy is driven by the distance measurement accuracy, which is in the order of centimeters for RIM sensors such as MESA® Imaging’s SwissRanger™ (Mesa® Imaging AG, 2011a) or PMDTechnologies’ CamCube (PMDTechnologies GmbH, 2007). Miniaturized size (including the modulated infrared light source), non-movable components and recording of kinematic processes (90 frames per second) are important advantages in contrast to laser scanners and stereovision systems. RIM cameras are suitable for detection and localization of objects/humans in indoor environments and can be used for simultaneous localization and mapping (SLAM), for example, onboard of autonomous Unmanned Vehicle Systems (UVS) (Weingarten et al., 2004; May et al., 2009).

Starting from this context, I will address four Research Questions within this thesis, which cover relative and absolute positioning in indoor environments.

Shortcomings and Research Questions

Ineffective 3D monitoring of industrial robots for decision making:

Industrial robots are able to execute given programs at high speed precisely with high repeatability (F. Lange and Hirzinger, 2006). Faultless functionality is important and performance to a specified speed and position accuracy has to be assured. For security reasons the working space of robots has to be monitored (ifm electronics, 2010). Shut-off mats or light barriers are used to minimize collision risk but cannot be adapted dynamically to the robots working space, which leads to larger security zones around the robot. An operator can analyze the present condition of the robot on his personal
experiences. However, due to frequent personal rotation, sufficient experience cannot be guaranteed (Blanc and Sjostrand, 2012). Visual camera control can offer dynamic adapted security zones as well as higher efficiency and better results in monitoring.

**Research Question 1:** Are ToF range camera measurements accurate enough to detect the path of an industrial robot in real-time to minimize potential risk of collision?

In populated environments robots and humans interact together. Simultaneous tracking and motion analysis leads to the research in Human-Computer Interaction (HCI). HCI as a “science of design” (Carroll, 1997) supports the interaction of human beings with technology. Cameras are mostly used as input devices for video communication. However, tracking of human body parts in front of an imaging sensor (e.g. Gorodnichy et al. (2002)) can be used for interaction and decision making.

**Research Question 2:** Can RIM sensors be used for HCI despite their limited range?

**Lack of absolute positioning in indoor environment:**
In the outdoor space location estimation and navigation is covered by Global Navigation Satellite Systems (GNSS). GNSS directly deliver absolute reference through their globally defined coordinate systems, like WGS84. In indoor environments, other solutions, for example ultra-wideband (e.g. Blankenbach and Norrdine (2010)), radio frequency identification (e.g. Kimaldi Electronics (2012)) or optical methods (e.g. Gorostiza et al. (2011)), can be considered due to the poor performance of GNSS.

**Research Question 3:** Is it possible to determine the absolute position of RIM sensors from prior known object’s location and geometry?

**Research Question 4:** Can RIM be used to generate 3D maps of indoor environments?

**Objectives**
The objective of this thesis is to present the usability of RIM in indoor navigation and location estimation tasks. The thesis is written in such a way that prior knowledge about ToF range cameras/RIM sensors is not an obligation, but basic knowledge in photogrammetry and/or computer vision and experience in dealing with image or point cloud processing will be helpful.

The suggested solutions rely entirely on open source technology and standardized code/GIS, aiming at enhancing the public usability of RIM for variable indoor environmental tasks. The introduction of IndoorGML should help to propagate geospatial standards that provide useful information and service exchanges in any applications that need to be geospatially enabled. Furthermore, standardized services are commonly trusted, widely accepted and highly valued.

**Overview of the Thesis**
This cumulative dissertation is structured in five chapters. Subsequent to this introductory chapter, the state-of-the-art and background of relevant research topics are detailed in Chapter 2.

The usability of ToF range cameras for object detection and monitoring (Research Question 1) as well as for HCI (Research Question 2) are addressed in Chapter 3. The SwissRanger™ 3000 (SR3000) (Oggier et al., 2005) is used as a static sensor to monitor
the scene from a single viewpoint. Its 16 bit gray value coded range images have been processed for object detection. Each grey value presents a coded distance where a grey color value of 65535 is equal to the non-ambiguity range of 7.5 m. The identified shortcomings in robot monitoring and decision-making based on 2.5D data processing are presented in Publication 1. Publication 2 focuses on multi-user operability in HCI. Following an overview of optical input devices and the presentation of other unique ideas, an example application using the SR3000 as input device is given, focusing on face and index finger detection.

During the last 5 years, several efficient 3D data processing algorithms have been made available, for example, the point cloud library (pcl). In Chapter 4 the potential of a RIM sensor as ego-motion system for indoor positioning (Research Question 3) is discussed. A case study based on a CityGML database model is presented in Publication 3. A model point cloud could be registered to acquired point clouds of the ToF range camera SR4000 (Mesa Imaging AG, 2011a) to calculate the camera’s position by resection. The advantage of integrating pcl algorithms for real-time 3D data processing is presented in Publication 4. Furthermore, the possibility to use the SR4000 simultaneously as mapping device (Research Question 4) is pointed out in Publication 5 when it was mounted on an Unmanned Vehicle System (UVS).

In Chapter 5 concluding remarks are presented and achievements are summarized. The scientific relevance of ToF cameras in indoor positioning is explained and an outlook on potential future activities in the field of RIM applications is given.
2 BACKGROUND

This section provides important historical context as well as the state-of-the-art of the most relevant topics of this research. It encompasses topics of photogrammetry and computer vision for robot monitoring and navigation as well as for Human-Computer Interaction (HCI) and indoor positioning through point cloud processing.

**Robot Monitoring for Online Collision Avoidance**

Online collision avoidance in unmanned factories is based on the Find-Path Problem (Brady et al., 1983) in multirobot systems illustrated in Fig. 1 and can be traced back to the late 1960’s. The robot’s motion can be considered as a path in its workspace. All movements must be accompanied by automatic online collision avoidance between all robots and all obstacles (Freund and Hoyer, 1986). The previous offline trajectory planning is not sufficient to totally avoid interferences with obstacles due to the fact that obstacles could move into the workspace (Chien et al., 1988). Therefore an online supervision of interacting robots is inevitable to handle situations, which cannot be preplanned. Information about the workspace depends on fast sensory input of the actual situation (Hoyer et al., 1994). The “Roboticians” (Laumond, 1998) of the 1980’s developed a variety of heuristic and approximate methods for collision avoidance.

![Fig. 1 Multi-robot system (KUKA Robot Group, 2006).](image)

With the growth of performance in low-cost computing and free-distributed software, efficient prototyping tools have been developed (Macchelli and Melchiorri, 2002). Kahane and Rosenfeld (2004) gave an overview of “robotic-assistant systems” and “highly autonomous systems”. Furthermore, the implementation of artificial vision, laser range sensors and machine vision devices in agricultural and construction tasks as well as in medical surgical operations is presented. A visual control system, like it is presented in Xu et al. (2006) makes autonomous robot manufacturing more efficient and reduces unnecessary production costs. Furthermore, the robot system needs to be

---

1 Background section contains excerpts from (Kohoutek, 2008), (Kohoutek, 2009), (Kohoutek et al., 2010), (Kohoutek, Mautz, et al., 2013) and (Kohoutek, Dröschel, et al., 2013) of this thesis.
independent from any a-priori knowledge about its position and orientation (Cefalu and Böhm, 2010). Recently machine vision is used to monitor the robot’s condition to detect malfunctions (Blanc and Sjostrand, 2012). Since 2007 the multi-camera system SafetyEye® (Pilz GmbH & Co. KG, 2007) is available on the market. While mounted on the roof the system covers an area of approximately 72 m² at its maximum range of 7.5 m but suffers from its dependency on background illumination.

The here presented work, Publication 1 “Analysis and processing of 3D-range-image-data for robot monitoring”, complements the previous works about robot monitoring to avoid collision in case of obstacles. The ToF range camera is placed in front of the robot and used as a Vision Based Protective Device (VBPD) (Hauke and Bömer, 2005). The robot will be detected in the scene and its path will be monitored. Any object or person that enters the hazardous area around the robot will cause a stop of the working process. The challenge is to provide enough safety of detection to limit the risk of collision. It is important to take worst-case scenarios into account due to the use of VBPDs. However, the robot’s condition is not monitored directly, because the accuracy of ToF range cameras may not be sufficient.

**Tracking of Body Parts for Human Computer Interaction**

In the 1990’s Virtual Reality (VR) (Rheingold, 1991) technology, the possibility to simulate the real-world with computers, became common in research. VR systems required the user to wear special gear like goggles, position tracker and gloves for gesture recognition. Arthur et al. (1993) presented “Fish tank virtual reality” and choose a head-tracking device (stereoscopic vision) and a high-resolution workstation monitor (Fig. 2) to overcome the use of head-mounted displays and the effect of separating the user from the real world.

![Fig. 2 Fish tank VR system (Arthur et al., 1993).](image)

Progress in hardware allowed using mid-range workstations for image processing to detect the user’s head position in real time. Estimating the user’s position and orientation towards the screen plus simple image processing techniques like template matching and background subtraction facilitated the head-tracking even in single camera approaches (Rekimoto, 1995). Facial recognition by characteristic feature points in frontal or profile view adapts the head-tracking to the user and can be used for training of neural networks and generic 3D facial modeling (Sarris et al., 2001).

Cipolla and Hollinghurst (1996) and Li et al. (1997) used stereovision respectively monocular vision to detect hand gesture without additional gloves. These approaches have been challenged by noise or bad lightning conditions. Applying active shape hand
models for estimation of the hands’ position and pose are presented in Hu et al. (2000) as another hand tracking and gesture recognition approach. Hahn et al. (2007) present a Multiocular Contracting Curve Density algorithm (MOCCD) to track the human hand-forearm limb. They chose a multi-camera system consisting of three calibrated cameras to avoid effects, which occur during pose estimation and tracking by the aperture problem. A 3D hand-forearm contour model consisting of five truncated cones is projected into each image. However, to compute the ground truth markers have to be labeled manually on the limb in the image. Furthermore, the contour model does not support gesture recognition of a single finger.

Correlating body templates with the camera frame makes the body tracking algorithm more robust (Betke et al., 2002). However, templates make the object detection process scale dependent, such that a limited number of predefined scales is applied, while people come in different shape, size and color (Koenig, 2007). Furthermore the tracking will be unsuccessful if the user rotates his/her head in angles that are not covered by the templates. Malassiotis and Strintzis (2005) and Srivastava et al. (2006) present a 3D head pose estimation approach based on active triangulation to track the tip of the user’s nose. The shape of the nose is visible during a large range of head rotation and unaffected by facial expressions.

Liu and Fujimura (2004) used ToF range cameras for hand gesture recognition. However, the authors used range data only for segmentation while the motion analysis is carried out based on 2D image data. Breuer (2005) used a calibrated ToF camera and principal component analysis for full 3D hand gesture recognition. Sabeti et al. (2008) combined color cameras with a ToF camera and realized scale independent face detection and tracking. Haker et al. (2009) fit a simple human body model into the acquired point cloud. With the estimation of a virtual touchscreen and their monocular computer vision approach for fingertip interaction Cheng and Takatsuka (2005) presented a concept for HCI that was used in an adapted form for Publication 2 “Multi-user vision interface based on Range Imaging”. The algorithm was later improved within the Bachelor thesis of Krähenbühl (2010). The achieved pointing interaction could be used to access object-based related information like it is presented in Haala and Böhm (2003).

Remark: In 2009 when Publication 2 was presented, Microsoft had not launched the Kinect™ yet. The presented research has to be seen in the context that real time body tracking in 3D was not a consumer market application. However, with the launch of Kinect™ interactive real-time human pose recognition as consumer hardware became reality. While Xia et al. (2011) proposed a model based approach to detect the human head Shotton et al. (2011) present a computational efficient and high accurate algorithm to predict 3D positions of body parts from a single depth image using a random forest classifier. The provided pose estimation algorithm was evaluated on real and synthetic data across differing body shapes and sizes. Furthermore, it handles self-occlusion and multiple people in the image. However, detected body parts roughly align with the body joints, which are inside the body. Moreover, the location of a joint cannot be estimated when its associated surface is occluded (Girshick et al., 2011). Taylor et al. (2012) used dense correspondences between vertices of an articulated mesh model and image pixels to estimate human pose in single depth and multi-view silhouette images. Besides the human pose and/or gesture a full body model can be calculated. It could be used for virtual fitting rooms or 3D avatars. Tong et al. (2012) as well as Böhm (2012) presented approaches using multiple natural user interfaces (NUI) (e.g. Kinect™) to acquire full human body models.
Camera Based Location Estimation in Indoor Environments

Navigation through unexplored environments is an important aspect of an autonomous mobile robot. There is a demand alternative indoor positioning methods due to the poor performance of GNSS within buildings and because of the lack of sufficiently accurate and recent maps of the interior. The main part of this thesis will focus on the investigation of Time-of-Flight (ToF) cameras for ego-motion determination in indoor environments.

Since the 1980’s several projects have been initialized with the goal to use mobile robots for the production of real world maps of indoor environments in an absolute coordinate system (Brooks, 1985). For example, the Hilare project (Giralt et al., 1984) deployed fix reference beacons for positioning and a directable laser range finder to produce models of the environment in an absolute coordinate system. The positioning method of Iyengar et al. (1986) does not need any initial previously learned model, but is able to continuously learn and apply results of previous acquisitions until the environment is fully described. Such incremental learning from mapping with range imaging devices was performed by Iijima and Yuta (1989) with the so-called “Yamabico M”, displayed in Fig. 3 and Freyberger et al. (1990) with their “Macrobe-robot”, which reduced the 3D range information to a 2D description moving space. Lux and Schaeffer (1991) presented real-time obstacle detection from range images acquired by a laser scanner for terrain analysis. In contrast to the approach of Reid and Brady (1992) their robot is able to modify its path in real-time to avoid collisions.

Fig. 3 Yamabico M (Iijima and Yuta, 1989)

El-Hakim et al. (1997) developed a multi-camera system consisting of eight CCD cameras that is using previously surveyed targets in a global reference system for location estimation. Besides the requirement of deploying dedicated markers, the natural scene can be used in a view-based approach. Objects with complicated structural elements are expected to be recognized in noisy images with the disadvantage of large memory and high computational costs for modeling and matching (Matsumoto et al., 2000). However, artificial and natural landmarks are distinct features with a fixed position in the absolute reference system, where the camera can be referenced to after a transformation has been carried out (Adorni et al., 2001). Böhm (2007) presented a point-based environmental model (PEM) to store prior knowledge of the scene in the reference coordinate system.

Bostelman et al. (2005) and Sheh et al. (2006) used ToF range cameras for rescue robotics to automatically generate textured 3D maps of almost unstructured environments. May et al. (2009) investigated simultaneous localization and mapping with a ToF camera, comparing Kanade-Lucas-Tomasi feature tracker (KLT) and Scale-
invariant feature transform (SIFT) algorithm to a depth-image based Iterative Closest Point (ICP) algorithm and the hybrid Efficient Second Order Minimization (ESM) technique. The drawback of ICP is its lead of convergence at an incorrect local minimum if the input point clouds are not already nearly aligned. Henry et al. (2010) used visual features in their RGB-D mapping approach to provide a first alignment without requiring initialization. Fusing RGB-D frames and ICP exploits the advantages of each to create 3D indoor maps. Their integration of color and depth information yields to robust frame matching and loop closure detection and generates 3D maps of indoor environments with low price RGB-D cameras. Sturm et al. (2011) proposed the first RGB-D dataset for benchmarking visual SLAM algorithms.

Publications 3 – 5 present an approach that does not require the deployment of any physical reference infrastructure inside buildings, which can be a requirement for a widespread implementation. The absolute position of the ToF camera is obtained with decimeter accuracy by a transformation from the camera coordinate system into the coordinate system of a Building Information Model (BIM). The advantages of the chosen spatio-semantic 3D CityGML (Gröger et al., 2008) database model are presented in Table 2.

| Classification criteria for 3D models of indoor space |
|-----------------|-----------------|
| **BIM** | **CityGML** |
| Creation process | Construction before building process | + Reconstruction after building process (e.g. laser scanning) |
| Geometric modeling | Constructive Solid Geometry | + Boundary Representation |
| | (Nagel et al., 2009) | (Nagel et al., 2009) |
| Semantic modeling | No | Yes |
| Data format | Computer graphics (VRML, X3D), CAAD/BIM-world (Industry Foundation Classes - IFC) | + GIS-world |

Geography Markup Language (GML) (Cox et al., 2002) is the standardized data storage format and interface of geographic information. The Standards Working Group (SWG) of the Open Geospatial Consortium (OGC) is developing a new GML for indoor navigation applications (Open Geospatial Consortium Inc., 2012a). The aim of the proposed IndoorGML is to provide a standard framework of interoperability between services and systems (Open Geospatial Consortium Inc., 2012b). IndoorGML will be used to describe the topology of buildings. The geometry however, will be described in existing standards like the CityGML. CityGML is a detailed Geographic Information System (GIS) model that contains information about geometry and semantics of indoor environments (Fig. 4) with all coordinates given in a global reference system.
Each object class, like `BuildingInstallation` or `BuildingFurniture`, has the attributes `class`, `function` and `usage` to capture semantic information. The range of possible models starts with small objects like barstools and goes up to huge objects like storage reservoirs. Objects, which are present several times in a room and have the same geometry, can be modeled in so-called implicit geometries. Their geometries can be saved in Cartesian coordinates in a database (e.g. CAD of the producer). CityGML links to those objects with an insertion point and a transformation matrix and there is no multi-storage of the same object geometry inside the CityGML.
Publication 1

Analysis and processing of 3D-Range-Image-data for robot monitoring

Tobias K. Kohoutek


(Author version; for typeset version please refer to the original journal article on http://versita.metapress.com/content/w7k6310gy4876246/fulltext.pdf)

Abstract   Industrial robots are commonly used for physically stressful jobs in complex environments. In any case collisions with heavy and high dynamic machines need to be prevented. For this reason the operational range has to be monitored precisely, reliably and meticulously. The advantage of the SwissRanger™ SR3000 is that it delivers intensity images and 3D-information simultaneously of the same scene that conveniently allows 3D-monitoring. Due to that fact automatic real time collision prevention within the robots working space is possible by working with 3D-coordinates.

Keywords: Range-Image, Motion Analysis, Object Tracking, Real-Time, Robot Monitoring, Security Zone, Optical Flow

1. Introduction

To record a moving object in 3D is possible using different methods. This paper shows the analysis and processing of images containing the local brightness and the distance for 25344 pixels. Those 3D-images are taken with the SwissRanger™ SR3000 using the ToF principle for measuring ranges. The miniaturized size (including the modulated infra-red light source of 54 LEDs), non-movable components and the recording of kinematic processes (30 frames/seconds) are important advantages in contrast to laser scanners.

Camera systems are rarely used for monitoring working processes of machine tools and industrial robots. Nowadays it is usual to use shut-off mats or light barriers to detect objects entering the security zone shown in Fig. 1.

Fig. 1 Robot working space (KUKA Robot Group, 2006).
2. Components

2.1 SwissRanger™ SR3000

This distance measuring camera, based on combining CMOS/CCD-technology, is developed by Centre Suisse d’Electronique et de Microtechnique SA (CSEM), Zurich Switzerland (R. Lange and Seitz, 2001). This camera is possible to acquire an amplitude image that shows the local brightness in gray values (16 Bit), and a range image for the distances in every pixel. The distances are coded into gray values (16 Bit) in the range image. The distance measurements are realized for each individual pixel by exploiting the ToF principle, working with a modulated infrared light source. Objects in a scene reflect the emitted light pulses back to the camera, where their precise time of arrival is measured at four points. In Oggier, Lehmann, et al. (2004) the phase map and finally a complete distance map can be acquired by detecting the phase delay between the emitted and the reflected signal in Fig. 2a. By sampling this signal the three unknown parameters of the modulated signal in Fig. 2b, the amplitude $A$, the offset $B$ and the phase $\phi$ can be determined by the equations (1) to (3).

![Fig. 2 Phase delay between the emitted and the reflected signal: a) phase delay (Zhang, 2004); b) modulated signal (Weingarten et al., 2004).](image)

$$A = \sqrt{(m3 - m1)^2 + (m4 - m2)^2}$$  \hspace{1cm} (1)

$$B = \frac{m1 + m2 + m3 + m4}{4}$$  \hspace{1cm} (2)

$$\phi = \arctan\left(\frac{m4 - m2}{m3 - m1}\right)$$  \hspace{1cm} (3)

With (4) and (5) the distance and the accuracy of the depth measurements $\Delta D$ can be calculated. $D_{\text{max}}$ represents the maximum unambiguous distance range of 7.50 m.

$$D = D_{\text{max}} \cdot \frac{\phi}{2\pi}$$  \hspace{1cm} (4)

$$\Delta D = \frac{D_{\text{max}} \sqrt{B}}{2A}$$  \hspace{1cm} (5)

2.2 LEGO® MINDSTORMS™ RIS 2.0

This robot was used to imitate the movements of an industrial robot. The Robotic Invention System (RIS) includes a RCX-Microcomputer, a USB-Infra-red-Transmitter and
the RIS Software (Windows™) as well as conventional LEGO® bricks and different sensors and motor types.

Interactive controlling is not possible with the RIS Software. Here the V2.1-Interface by Berger (2005) programmed in Microsoft® Visual C++ was adapted. Due to that fact the RCX-Microcomputer works only as an interface between the motors/sensors of the robot model and a desktop computer.

3. Analysis and Processing of 3D-Image-Information

3.1. Robot Detection

With the experimental setup shown in Fig. 3 two different kinds of segmentation could be realized by using Microsoft® Visual C++ and the Open Source Computer Vision Library OpenCV (OpenCV dev team, 2011). First method of segmentation is based on detection by using a background image. Here the difference image of the background image and the first image with the robot displays as result only the robot itself. It is not possible to work with a background image for the second method if segmentation is used.

Thresholding, morphological operations and edge detection are utilized algorithms for the segmentation. In every case the initial image presents the mean over 25 images to reduce the noise level. For \( N \) images the Signal to Noise Ratio (SNR) in a mean image is \( 1/\sqrt{N} \) (Jähne, 2005). The range image was used for the edge detection, due to the fact that the scattering of the pixel values is bigger compared to the amplitude image. In Fig. 4 different edge detection algorithms were used like Sobel- (Fig. 4a and 4b), LaPlace- (Fig. 4c) and Canny-algorithm (Fig. 4d).
The best result, a binary image, was given by the Canny-algorithm. Because more edges can be detected than actually are on the robot itself, a threshold was set up to remove those outlines with a fewer number of pixels as those at the robot contour. This step is followed by a dilation operation to fill the gaps in the contour. In a last step the robot structure is filled out white and features become marked is the binary image using a fixed raster over the white area shown in Fig. 5.

In the following steps only features at the robot's contour are required. A simple neighborhood operation is used to minimize the features by proofing every feature and its four neighbors (up, down, left and right) whether they belong to the robot (white) or to the background (black). A feature is an edge feature when three neighbors belong to the robot and the fourth to the background. It also counts as an edge feature when only two neighbors belong to the robot, but both should not be on opposite sides. In this case, the feature is a corner feature. (Fig. 6) The result is a new image that contains only features at the robot contour.
3.2. Virtual Security Zones

For monitoring and collision prevention a virtual security zone around the robot is created. There are two different shapes shown in Fig. 7, where the first one is similar to a cube around the robot and called “static security zone”. The second shape fits much better to the robot shape and is called “dynamic security zone” because it will do the same movements like the robot.

In order to create the static zone, three special points at the robot contour have to be known. These points are the furthest features to the left, right and top of the robot edges. Knowing the X- and Y-values in the image it is possible to append the virtual cube to these features around the robot. The cube thickness of the cube is correlated to the robot’s velocity and describes the security area. The stopping distance increases with higher speeds so consequently as faster the robot moves as thicker the cube has to be. It is obvious that much more image space than necessary is controlled or not usable for other image operations by using such a “static” box. Due to that fact the dynamic zone was created. It fits much better to the robot contour because the shape is created by all edge detected features. Now it is important to find out whether a feature is on the left or the right side of the robot, or at the top or the ground. Therefore a more sophisticated neighborhood operation was used.

For all features (shown in green) it is known on which side at the robot contour they are. Using that information, they are shifted in radial direction for five pixels on a line from the focal point (yellow) to the image edge. These virtual inner boarder points (blue) were shifted a second time in the same direction for eight more pixels in order to create the outer barrier (red). After definition and preparation of the security zones in the 2-dimensional image space a concept for the 3D robot working space was generated. As seen in the top view in Fig. 8, the monitored working space was split into three areas (front, robot depth, rear).
The idea is to detect other objects entering the chosen security zone in three different distances. If there is an object entering the security zone between the camera and the robot (front), the robot has to stop immediately. In that case the robot would be in the so-called phantom space (detailed information in Franz (2005)) of the other object and it is not possible anymore to guarantee a risk free working of the robot. The following controlled zone is called “robot depth” and is as depth as the robot itself. Directly behind the robot there is again a phantom space, which is not controllable. It seems that objects entering the security zone from behind cannot be detected. But if they come nearer to the robot they “grow” in the image space and the gray values in amplitude image and range image will change.

The conditions that have to be fulfilled to stop the robot’s work are simple. An object is entering the security zone if there are changes in the gray values of the amplitude image as well as changes in the range image. There are three different distance thresholds for controlling the range image in the distances “front”, “robot depth” and “rear”.

4. Robot Monitoring/ Tracking and Motion Analysis

Motion in image sequences is always associated with changes between two images. By subtracting these two images all differences become visible. It is important to know that gray value changes are not always related to object motions. Also changes of the light source or camera position generate such differences in the gray values as can be seen in Fig. 9.

The results are the motion field and the optical flow, where the motion field describes the real motion of an object in the 3D-scene projected onto the image plane (Jähne, 2005). The optical flow (Horn and Schunk, 1981) is defined by the flow of gray values in the image. The existence of a constant light source and a pixel neighborhood, that moves similar to the center pixel (Fig. 10), during the exposures are pre-conditions for using the optical flow.

This displacement vector field is the projected 3D physical motion filed to the image plane and provides information about the arrangement and the changes of objects. The gray value’s “flow” over the image plane is equivalent to the flow of volume elements in liquids or gases. Motion is closely related to spatial and temporal gray
value changes. That is why only the component of the displacement vector, which is normal to an edge, can be determined while the component parallel remains unknown. That problem is called aperture problem and applies only local operators. In Fig. 11a an unambiguous determination is only possible for an object corner that lies within the operator mask. The aperture problem is a special case of the correspondence problem, because distinguishing different points of an edge is not possible.

A solution for that problem is shown in Fig. 11b by an image pyramid that reduces the resolution of an image gradually. While the original image \( I \) is the 0th pyramid level the resolution and size of the following \( I_L \) decreases by a factor of two. Using pyramids into a magnitude of local neighborhood operations scales large pictures down. Smaller image size neighborhood operations made in the upper level of a pyramid can be performed more efficiently than for finer scales. Important image features build the basis for the hierarchical image classification. Image pyramids express a high robustness and a good local accuracy.

![Fig. 11 a) Aperture problem, b) Image pyramid (Siebold, 2004).](image)

During the working process of the robot the motion of all object features will be calculated by the optical flow with sub-millimeter accuracy. Their displacements will be added to their corresponding points of the dynamic security zone and so this zone follows the same movements as the robot. During the real time robot monitoring (30 frames/second) the security zone will be controlled in amplitude image and range image. The robot will stop when there is a detected gray value changing in both images at the same feature and its direct neighborhood. If an object is detected, the robot will stop in time to avoid a collision and the object will also be marked and displayed in the image displayed in Fig. 12.

![Fig. 12 Detected object.](image)
5. Conclusion

The determination of a robot working space and its automatic real-time monitoring was carried out successfully. The results for the analysis and processing of the 3D-images are reliable. Multipath effects, temperature, distortion and effects of the objective influence the distance measurement. The data for absolute and relative accuracy of the recorded objects will become reliable after a camera calibration of the SwissRanger™ SR3000.
Multi-user vision interface based on Range Imaging

Tobias K. Kohoutek

Proc. SPIE 7341, Visual Information Processing XVIII, 73410M (April 27, 2009); DOI: 10.1117/12.819385

(Author version; for typeset version please refer to the original conference proceeding on http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1338961)

Abstract For common computer interaction the mouse is established as a standard device. The recognition of freehand 3D-interaction has already been implemented by detecting the fingertips and the eyes of the user. This application is based on the stereo-photogrammetry approach with two webcams. Attempts with a single webcam have been performed as well to quit the synchronization of two video streams. Using the range-imaging technology the user can move in front of the display from 30 cm up to the maximal ranging distance that is supported by the camera. The body, especially head and hand, can be detected in 3D within the operating range and an additional gesture-analysis tool is able to interpret the commands of the user. With this approach, the computer mouse is not needed anymore. The main topic of this paper is the multi-user interaction. Operating the computer at the same time with several users is not supported by the actual operating systems. The simultaneous detection of several users and their hands in 3D was achieved. A fast switching between the users to control the computer in turns is explained.

Keywords: 3D-tracking, Real-time, Range Imaging, Human Computer Interaction, Multi-user interface

1. Introduction

Computer screens and displays have become larger in the past; also digital projectors have been used much more often. This has inspired research to use other input devices then keyboard and mouse. Since the early 1990’s much research has been carried out in Human Computer Interaction (HCI). According to Erol et al. (2005) the human hand is the most effective interaction tool for HCI. Gorodnichy et al. (2002) carried out research for completely hands-free interaction that requires motion measurement and tracking of various human body parts. Most research in computer vision uses stereo photogrammetric approaches, for example Cipolla and Hollinghurst (1996); Grätz et al. (2004); Erol et al. (2005) or the single camera (webcam, digital video cam) approach, for example Gorodnichy et al. (2002); Cheng and Takatsuka (2005) or Lertrusdachakul et al. (2005). On the other hand there are some less known methods. Oka et al. (2002) used an infrared camera to detect areas close to the temperature of the human body. Banker et al. (2007) designed a 3D computer mouse while using an ultrasonic transmitter. The active triangulation principle by Malassiotis and Strintzis (2005) is based on the coded light approach for 3D data acquisition.

Besides the detection of the user’s head a detection of the user’s “pointing device”, the hand/fingertip, is required in many applications. Nickel and Stiefelhagen (2003)
explained the importance of pointing device detections with: “Humans perform the arm towards a pointing target in the communication with others to mark a specific object, location or direction.” The 3D geometry and the gesture in natural arm movements have to be detected and the pointing direction has to be estimated. Consequently several problems have to be solved to implement a virtual mouse.

The acquisition and distance measurement is realized for each individual pixel by a ToF range camera exploiting the ToF principle. Objects in a scene reflect the emitted light pulses back to the camera, where their time of arrival is measured. A ToF range camera combines the benefits of single and multi-camera systems. RIM enables to measure the user’s position in 3D using only one camera.

The second step is to create a multi-user interface by upgrading the existing image processing algorithms from a one user to a multi user system independent from the user’s movements. However, it is not possible to operate with multi users at the same time due to limitations of the operating systems (MS Windows™, Mac OS, Linux). Only one pointing device can be implemented. The advantage is to switch among the several users who were detected during image processing.

2. Range Imaging

While using a single camera, the distance between the user and the camera/display is determined easily by the user’s head size (Cheng and Takatsuka, 2006). In contrast, approximation of the distance between the user’s hand and the screen/camera remains a different task. Stereo or multi vision photogrammetry acquires a 3D-model but analyses and matching of multiple video streams is needed. Common methods to measure 3D are stereo triangulation, sheet of light triangulation, structured light projection or interferometry. RIM however uses the ToF principle.

There are two different ways to measure distances using ToF. One method is the time measurement of a laser pulse, which is reflected at an object. Laser scanners use this principle. Another type of ToF is working with a modulated infrared light source to measure the phase delay as it is done in this work. The accuracy of the distance measurement is the limiting factor in HCI for computer vision.

ToF cameras provide real time distance data at video frame rates up to 50 frames per second. They acquire an amplitude image and a range image. The local brightness as well as the distances for every pixel is coded in 16 bit gray values. For each pixel the distance is measured directly by calculating the phase shift between the emitted and reflected signal. The phase map and finally a complete distance map can be acquired by detecting the phase delay between both signals (Oggier, Kaufmann, et al., 2004) as it is shown in Fig. 1a. By sampling this signal the three unknown parameters of the modulated signal in Fig.1b, the amplitude $A$, the offset $I$ and the phase $\phi$ can be determined by the following equations (1) to (3), where $m1 \ldots m4$ are the measured phase delays.
Fig. 1 Phase delay between the emitted and the reflected signal:
a) phase delay (Zhang, 2004); b) modulated signal (Weingarten et al., 2004).

\[ \varphi = \arctan \left( \frac{m_4 - m_2}{m_3 - m_1} \right) \]  
\[ A = \frac{\sqrt{(m_3 - m_1)^2 + (m_4 - m_2)^2}}{2} \]  
\[ l = \frac{m_1 + m_2 + m_3 + m_4}{4} \]  

With (4) the distance \( D \) and the achievable depth resolution \( \Delta D \) can be calculated.

\[ D = D_{\text{max}} \times \frac{\varphi}{2\pi} \text{ with } \Delta D = \frac{D_{\text{max}} \sqrt{l}}{2A} \]  

The maximal operating range of the camera is represented by \( D_{\text{max}} \). Every object, which is further away, will be shown in a wrong gray value due to the repetition of the modulated signal. State-of-the-art ToF cameras operate with a wavelength of 850 - 870 nm and a modulated frequency of 25...30 MHz. The current array size dithers between 60 x 60 up to 205 x 205 pixels.

3. Virtual Touchscreen

In a first step the camera position has to be related to a large screen. Usually, the camera is placed on the top of the screen. Fig. 2 shows that the view frustum between user and display is flexible while the user moves in the operating distance of the ToF range camera. While using this technology, the frustum will be adjusted to the user's body and get re-adjusted with the user's movement in real time. A virtual touch screen appears in front of the user by pointing to the display. The virtual screen will always be within the frustum and is adjusted to the user's arm length.

Fig. 2 View frustum and virtual touchscreen (modified from Cheng and Takatsuka (2006)).
3.1 Hand-Head Line Model

People tend to look towards an object and use one finger to point in the direction of the object when they show it to somebody else. The extension of the finger and the eyes are in the line of gaze with the object, shown in Fig. 3. The same happens here with display and user. All objects on the screen are not touchable from the distance but at the fingertip the virtual display will be placed (Cheng and Takatsuka, 2006), due to checking the range image for the closest point to the camera by histogram evaluation. Remember, that the distance is stored in a 16 bit gray value image. While the fingertip will always be the closest point to the camera, the gray value will be an extreme of the image values. The minimal gray value presents the fingertip of the user.

Pointing to the four corners of the screen is one possibility to calibrate the system. This way the view frustum becomes adjusted to the user’s arm length. Because of the RIM technology the user is now able to move forward or backward, left and right in front of the display. By pointing towards the display the mouse pointer is now driven by the fingertip. The fingertip interaction, left click, right click and mouse wheel scrolling are parts of the gesture recognition but are beyond the scope of this work. There are several papers dealing with gesture recognition, for example Hu et al. (2000); Nickel and Stiefelhagen (2003); Erol et al. (2005); Breuer (2005) or Argyros and Lourakis (2006).

4. Multi-User Interface

As previously mentioned, detection and tracking of a single user in front of a camera has been achieved already in many different works. The challenge here is to generate multi-user interface functionality by 3D image processing. The idea for multiple users on a computer screen originates from a blackboard where several users can write or draw simultaneously. This work will show the possibility of detecting and tracking multiple users in 3D. The used ToF range camera SR4000 is developed by MESA® Imaging (Mesa® Imaging AG, 2009). Its pixel array size is $176 \times 144$, working with a wavelength of 850 nm and a modulated frequency of 30 MHz. The operating range with standard settings is 0.3 to 5.0 meters.

The complete source code is written in MS Visual Studio 2008 C++ enhanced to the offered library files for camera acquisition and image processing. In a first step a real time face tracking with scale invariance could be realized. Therefore the Haar-face detection method was used. The Haar-like feature classifier is given by the Open Source Computer Vision Library OpenCV (OpenCV dev team, 2011). This algorithm uses a face template to match it with the users’ faces. The mimic of the users has no influence to
the template matching. Not yet implemented is a real time tracking of the eyes such as provided by Savas (2005). Tracking the eyes is important when the users are close to the display (<1.5 m). In this case, the hand-head line is established between eyes and hand. Actually, the mid-eye position and the fingertip build the line of gaze.

Normally one computer is operated by a single user who only needs one keyboard and one mouse to control the system. Currently a second pointing device is not supported by the operating systems of today. A second courser cannot appear on the screen by simply connecting a second mouse with the computer. It is only possible to move one screen courser with both devices. In a multi-user application there are at least two users (A and B). The face of user A has to be detected as well as the one of user B. Also both mid-eye positions have to be calculated. Each position has its own view frustum towards the display. By detecting the fingertips of A and B there will be four lines of gaze. Of course, only two of them are correct. Fingertip of user A has to be connected with the mid-eye position of user A. The same has to be done for user B. If fingertip A is combined with mid-eye B or vice versa then a wrong object in- or outside the view frustums is pointed at. The bodies of the users give the correct combination between the hands and the heads of users A and B. A fingertip is always connected to a hand, which is connected to an arm that is connected to the body where the user’s head is on top. A 3D body-tracking, like it is presented in (Guillaume, 2006), is necessary to fix the users lines of gaze. All necessary information can be obtained from the range image.

The screen courser is calculated where the line of gaze of a user intersects with the display. Assuming that user A is controlling the operating system, the arm and fingertip of user B should not be raised. If user B wants to control the system now he has to rise up his arm not before user A brought his arm down. This order is crucial to pass the pointing device on user B. The reason for that order is that both users do not have to stay in the same distance to the display. If user A would be closer to the camera he would stay in front of user B, even when user B would lift his arm. The fingertip detection is based on the definition of distance regions. The minimal distance obtained by the minimal gray value in the range image represents the finger of the user. If user A is closer to the display its body represents the minimal gray value. To solve this problem a threshold is used. A body is represented by more pixels then a hand or fingertip. The solution is to count the pixels close to the camera. If there are more pixels than the defined threshold then the detected object cannot be the fingertip. The algorithm has to start again and search for the next points close to the camera. While head and hand of user B are detected he will be the new system operator.

5. Results

It can be observed from Fig. 4 that the ToF range camera captures the distance information for each pixel. The closest point to the camera is the pointing fingertip towards the screen. The detection algorithm was successful.
The face detection performs best between 130 cm to 400 cm. If the user is too close to the camera too much reflected light gets on the sensor, and if too far away, the face is too small for the detection. Fig. 5 shows, that the face detection algorithm lost the tracking of a head when the user turned around because the used algorithm is not rotation invariant and the template for the matching requires a face to be oriented towards the front. This phenomenon occurred for larger distances even under small rotations. Solving this problem will be a part of future works.

6. Conclusion

A set of experiments with real-time head and hand tracking was performed. Volunteers were asked to move their heads in a natural way and point one by one towards the display. The fingertip was found successfully for every user. The screen courser was set to the position where the line of gaze intersects with the display.

Another challenge is the gesture recognition with the ToF range camera. For instance, a fast forward-backward motion could be implemented as a click. In order to enable this, the distance measurement accuracy in the range image has to be increased (Kahlmann et al., 2006) to adjust the view frustum more accurately. However, a simultaneous
interaction of multiple users could not be realized because of the limitation of pointing devices by the operating systems. A fast switching between users has been found to be the only possibility to allow several users operating the same computer simultaneously.
4 INDOOR POSITIONING USING A TOF RANGE CAMERA

Real-time indoor positioning using Range Imaging sensors

Tobias K. Kohoutek, Rainer Mautz, Andreas Donaubauer

Proc. SPIE 7724, Real-Time Image and Video Processing 2010, 77240K (May 04, 2010); DOI: 10.1117/12.853688

(Author version; for typeset version please refer to the original conference proceeding on http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1345883)

Abstract: This paper considers a novel indoor positioning method that is currently under development at the ETH Zurich. The method relies on a digital spatio-semantic interior building model CityGML and a Range Imaging sensor. In contrast to common indoor positioning approaches, the procedure presented here does not require local physical reference infrastructure, such as WLAN hot spots or reference markers.

Keywords: Range Imaging, CityGML, Indoor Navigation, Building Information Modeling

1. Introduction

The development of indoor positioning techniques is booming at the moment. For industrial applications such as automation, warehousing and logistics there is a significant demand for systems that have the capability to determine the 3D location of objects in indoor environments without the requirement of physically deployed infrastructure. In particular, tracking of persons in indoor environments has become vital during firefighting operations, in hospitals and in homes for vulnerable people especially vision impaired or elderly people.

In contrast to the majority of indoor positioning methods, the novel method described in this paper does not require any physical reference infrastructure (e.g. Wi-Fi hot spots) inside buildings, which can be a decisive advantage in respect to other methods at the same level of accuracy. Instead of a locally deployed reference infrastructure, the method relies on a digital spatio-semantic interior building model, based on the CityGML scheme (DeSouza and Kak, 2002). The paper describes the status of current research conducted at the ETH Zurich. The research questions behind this work are:

What should be the ideal concept for an indoor positioning method that is based on RIM and semantically rich geospatial data (CityGML) instead of relying on physically deployed infrastructure (e.g. Wi-Fi access points)?

Which level of accuracy can be achieved and what application scenario has the method?
What are the strengths and drawbacks of the method compared to methods that rely on physical reference infrastructure and do not make use of a spatio-semantic model for indoor environments?

2. CityGML as a Method for Modeling Indoor Environments

2.1 Overview of Approaches for Interior Building Models

Different disciplines such as Computer Aided Architectural Design (CAAD) / Building Information Modeling (BIM), Computer Graphics and Geographic Information Systems (GIS) are dealing with three-dimensional interior building models with each of the disciplines having developed a different design that has been tailored to their application. The following criteria can be used for classifying these approaches:

1. Creation process: CAAD/BIM models are normally generated during the planning process of a building before the construction phase and therefore represent the building as it was designed before it has been built. Computer Graphic Models (e.g. for virtual tours) and 3D GIS (e.g. for Computer Aided Facility Management applications) represent the state of the building after its completion. These models are often derived from measurements taken inside the building. The models that were created during the planning phase of a building often contain information that is no longer visible after completion of the building (e.g. columns, which are integrated in walls, cables and pipes or beams under suspended ceilings). In contrast, models that have been created after completion of the building normally contain only the visible parts of the building interior.

2. Geometric modeling: There are two paradigms for modeling 3D vector geometry (Kolbe and Plümer, 2004). In the Boundary Representation (b-rep) paradigm, where a room is represented by its bounding surfaces and the boundaries being lines or curves. The 3D coordinates for each vertex of a boundary are stored individually. B-rep is the common modeling paradigm for vector geometries in GIS. CAAD/BIM models often apply the Constructive Solid Geometry (CSG) paradigm. In CSG, complex 3D solids are derived by combining 3D primitives such as cuboids, spheres and cylinders with the help of Boolean operators union, difference or intersection. The parameters of the primitives and coordinates of insertion points are stored.

3. Semantic modeling: Models of indoor environments that are used for visualization purposes only, usually contain little explicitly modeled semantic information. Those models focus on the geometry and the data that is used for controlling the graphical representation of geometry such as textures. In contrast, models from the CAAD or BIM domain provide a large amount of semantic information as they rely on detailed semantic information for each construction component of a building. For example, the standard IFC defines more than 600 semantic object classes in a building model (Gröger et al., 2008).

Depending on the application different data formats are used for the codes and data transfer. In applications for the purpose of visualization, VRML or X3D formats are common. The standard IFC format (buildingSMART, 2008) fulfills the requirements for applications in architecture and construction. The standard CityGML that is designed for GIS applications and used in our application is discussed in more detail below.
2.2. The CityGML Indoor Space Model

The standard CityGML (DeSouza and Kak, 2002) defines a data model and an XML data format for 3D city and topography models. CityGML defines several Levels of Detail (LoD) with the highest LoD 4 having the capability for modeling the interior of buildings. In particular for the purpose of indoor modeling, the semantic model provides an object class ‘Room’ that can capture semantic data and contains the attributes class for the classification of rooms, function for the intended use and usage for the current use of the room such as living room or office.

An object of the class ‘Room’ can be associated with its geometry in two different ways. One way of defining the outer shell of a room is to establish a link to a geometric object of type Solid or MultiSurface (both types are defined by the GML 2.1.2 specification (Cox et al., 2002)). Alternatively, the outer shell can be decomposed into semantic objects of the types InteriorWallSurface, CeilingSurface and FloorSurface. These semantic objects refer to geometric objects of type MultiSurface. Openings in the outer shell of a room can be modeled by the use of the object classes Window and Door that can belong to one or two InteriorWallSurfaces. This data structure can be used to express topological relationships between rooms. Permanent fixed objects belonging to a room (e.g. radiators, columns, beams) can be modeled using the semantic object class IntBuildingInstallation. In order to model the mobile components of a room such as desks and chairs, the object class BuildingFurniture can be used. IntBuildingInstallation and BuildingFurniture provide the attributes class and function and usage for semantically describing the objects.

The geometry of these fixed installed objects can be defined by the standard GML 3.1.1. In addition, the geometries of the variable components of a room can be modeled using the so-called implicit geometries. Hereby the shape of an object is stored only once in the library even if multiple objects of the same shape are present (e.g. pieces of furniture). For each occurrence of such an object, only the local coordinates of an insertion point and a transformation matrix are stored. They are then linked to the geometry that is captured in the CityGML. Using this mechanism, the model could have a direct link to the 3D-CAD-drawings of pieces of furniture in the manufacturer’s catalog. Fig. 1 shows the indoor model of a room using the semantic classification of CityGML.

Fig. 1 ETH Zurich lecture room HIL C71.3 in CityGML (Donaubauer et al., 2010).
3. Range Imaging as a Method for the Purpose of Indoor-Positioning

Positioning awareness and navigation capabilities have become increasingly important for many applications in all environments. Global Navigation Satellite Systems (GNSS) and surveying total stations are able to cover the positioning requirements for outdoor applications. However, these systems have weaknesses indoors. Due to the importance to deliver position in indoor environments, various alternative approaches such as those exploiting signal strengths indicators, intermodal ranges by ToF for trilateration or angular measurements for triangulation have been developed with not yet satisfying performance. A drawback of these approaches is the often missing connection to the global geodetic coordinate reference that is used outdoors.

Fig. 2 gives an overview of current positioning systems according to their specific coordinate accuracies and coverage. On the left side of the graphic, the high precision systems used for applications in industrial metrology are shown. The drawback of systems with positioning capabilities in sub-millimeter precision is the small coverage and the requirements for expensive local installations. In contrast, inexpensive systems that have been developed for low-accuracy applications are shown on the right side of Fig. 2. These systems exploit the Received Signal Strength Indicators (RSSI) in order to obtain positioning capabilities within meter accuracy or to resolve the position of a device within room level.

For many applications the requirements of positioning accuracy is within millimeters to centimeters. This level of accuracy can be reached with geodetic methods such as total stations or rotational lasers. In recent years, network based methods that obtain range or time of flight measurements between the network nodes have become significant for applications at decimeter level accuracy. The measured distances can be used to determine the 3D position of a device by spatial resection or multilateration.

In particular vision based methods have become an interesting alternative for indoor positioning and navigation. The performance, the size and the speed of CCD and CMOS sensors have grown rapidly in the last view years. The computing speed and the algorithms for feature recognition in images obtained by digital cameras have reached
an unprecedented performance. Optical methods use digital images to recognize points, codes, features and objects in order to determine their image coordinates. These 2D image coordinates can be transformed into the reference coordinate system with the goal to determine the camera position by spatial resection.

Manufactures of optical indoor positioning systems such as AICON 3D Systems GmbH (2009) with their camera system ProCam offer high precision positioning systems in the range of \(\frac{1}{10}\) mm for applications in optical metrology, in particular for surface inspection or reverse engineering. However, these systems require the installation of an active field of reference points and a prior calibration of the system.

One approach to avoid the dependency on a reference field is the project CLIPS (Camera and Laser Indoor Positioning System) launched by Mautz (2010). The system uses the fundamentals of stereo photogrammetry, where the position and the rotation of a camera relative to another camera are derived. But instead of using a second camera, it is replaced with a device called laser-hedgehog that projects well distributed laser spots as flexible reference points on the ceiling, walls and furnishings in any indoor environment. The projection creates a flexible field of reference points that can be observed by the real digital camera. Tilch and Mautz (2010) have shown that the CLIPS camera could be located with an accuracy of sub-millimeter.

Matsumoto et al. (2000) and DeSouza and Kak (2002) present an overview of attempts that have been made to exploit view- or map-based indoor positioning systems for mobile robot navigation in indoor environments.

The method based on RIM that is subject of this article belongs to the optical map-based indoor positioning systems. In contrast to traditional optical sensors, the range image does not reflect the brightness of the objects in the scene, but the distance of these objects to the ToF camera. The expected 3D position accuracy for objects seen by a ToF camera (in terms of a 1-\(\sigma\) standard deviation) is 1 cm for distances of 2 m and 1 dm for distances of 10 m. The largest error budget contributes the low-accuracy distance measurement. According to the manufacturer the ranging accuracy of the current model SR4000 is 1.5 cm for objects in 8 m distance at a level of reflectivity of 100 %.

RIM can be particularly used for the purpose of indoor positioning, because in contrast to the other methods mentioned above RIM can exploit semantic 3D geoinformation models. The methods will be detailed in Chapter 4.

3.2. Range Imaging

Common methods to measure 3D point clouds are stereo triangulation, sheet of light triangulation, structured light projection or interferometry. In contrast to these techniques, RIM uses the ToF principle. There are two different ways to measure distances using ToF (Kahlmann and Ingensand, 2005). One method is the time of flight measurement of a laser pulse, which is reflected at an object. Most laser scanners use this principle. The other method that is applied by the sensor used in this research, measures the phase delay of a modulated infrared light signal. ToF cameras (Fig. 3) provide real time distance observations at video frame rates up to 50 frames per second.
4 Positioning Using Range Imaging and CityGML

4.1. Overview of the Proposed Method

The presented positioning method consists of two main components, RIM sensor (described in Section 3) and a semantic-geometric 3D database that is modeled in CityGML (details given in Section 2). Details on the processing, data storage, analyzing and data transfer remain critical for the realization of the system, but cannot be given within the scope of this paper. A description of the method is given below.

4.2. Room Identification Through Object Detection and Processing Using the CityGML Database

This first step has the goal to identify the room in the CityGML database, where the camera is located. The detection and identification of objects is the key part of this step, which can be achieved from the amplitude image of the range imager that is similar to a grayscale optical image of the scene. In order to identify the objects such as chairs, tables, etc., known or “learned” primitives, features and image samples from the libraries (that are described in Section 2.2) are matched with the image data from the camera. The detected object properties such as the size, geometry or quantity of a certain object are the main criteria for the comparison with the database. This way, the unknown camera position can be reduced to a small number of possible rooms. By detecting distinct properties the room can be identified uniquely and additional semantic and geographic information can be extracted from the 3D geo-database. Fig. 4 shows the comparison between an observed 3D point cloud from the RIM sensor and a form primitive of a database model.

4.3. Accurate Positioning from Distance Measurements

The second step of camera localization is the precise positioning part, described in this section. This step compares and transforms the local coordinates of the objects that have been recognized by the camera into the reference coordinate system of the database. The reference points for the transformation are the corners of the room, vertices of doors, windows and other fixed installation or furniture. The accuracy of the objects in CityGML should be at centimeter level\(^2\) and should lead to position determination of the camera with centimeter-accuracy using a least squares adjustment with a redundant number of reference points to determine the 3D camera position. One requirement for the camera is that its inner orientation has been determined previously. The outer camera orientation and position are determined by a technique that combines trilateration (based on the distance measurements) and spatial resection (based on the image coordinates that are translated into horizontal and vertical angles). If there has been an ambiguous solution in the identification at room level in step 1, the precise positioning step has the potential to disambiguate and deliver only one unique solution for the correct room. Further research needs to be

---

\(^2\) The Standard CityGML at level of detail 4 (LoD 4) defines a horizontal and vertical accuracy of 0.2 m.
investigated with the goal to exploit the semantic information that the CityGML database holds.

![Object comparison between a range image (left) and form primitives from database (right).](image)

**Fig. 4** Object comparison between a range image (left) and form primitives from database (right).

### 4.4. Opportunities and Limits of the Proposed Method

Kinematic acquisition of 3D-coordinates in real-time allow for efficient recognition of rooms and the position of objects in those rooms in relation to a given model. The identification of objects can be trained with the help of neuronal networks. Currently, the relatively small distance measurement range limits the proposed method. Modern RIM sensors are able to measure distances unambiguously between 5 – 10 m at an accuracy level of centimeters. The ambiguity problem arises from the frequency of the modulated signal of the RIM sensor. For example, the SR4000 camera has a unique distance range of 5 m, i.e. an object in 6 m distance from the camera could also be in 1 m or 11 m distance. The ambiguity problem can only be solved with additional prior information.

Another problem pose the so-called mixed pixels that are obtained when the signal from the ToF camera hits an edge of an object. Then, the signal is partially reflected at the foreground, but also reflected at the background. Both signal parts arrive at a single CCD element. As a result, the values of the mixed pixels consist of an average between the foreground and background distance. In the point cloud, these pixels appear as single unconnected points that seem to float in the air and that do not belong to any object. This is also a common problem in terrestrial laser scanning. Note that systematic optical influences such as focussing, vignetting\(^3\) and aberation\(^4\) must also be determined by a prior calibration and need to be corrected accordingly.

---

\(^3\) Vignetting is the effect reduction in an image's brightness or saturation at the periphery compared to the image center.

\(^4\) Aberation is the optical distortion that leads to a local variation in scale due to geometric errors of an optical system.
CityGML seems to be an appropriate basis for the positioning method for the following reasons:

1. The coherent spatio-semantic model with its high level of detail and the positioning method are complementing one another. Range-imaging and other imaging methods used for positioning can capture visible objects. Also in CityGML, only visible objects are modeled. Partially visible objects are split in such a way that only their visible part is modeled. This approach is different from semantic interior building models from the CAAD/BIM domain, which contain a high amount of objects, which are invisible (like cables or pipes) or just partially visible (like beams spanning several rooms) and therefore do not fit the RIM method very well.

2. For CityGML there is a standardized Web access interface, the OGC WFS (Vretanos, 2005), which provides access to data and associated operations (e.g. geometric and thematic filtering of data).

3. In contrast to models from the CAAD/BIM domain, CityGML supports any geodetic reference system. If an interior building model is already in the geodetic reference system, the coupling of indoor and outdoor positioning methods is straightforward.

Further practical tests and an assessment will show whether our optimistic view for the use of CityGML turns out to be justified or whether CityGML needs to be further extended as shown in Bleifuß et al. (2009) in the facility management domain.

5 Outlook

First steps towards a realization of the proposed indoor positioning method have been carried out with a ToF camera. In parallel, parts of an office building at the ETH Zurich have been modeled in CityGML. The next steps are the implementation of the coarse and the fine positioning method. These methods need to be tested in order to have an answer to the questions that have been raised in the beginning of this paper.
Abstract: We present a novel approach for autonomous location estimation and navigation in indoor environments using range images and prior scene knowledge from a GIS database (CityGML). What makes this task challenging is the arbitrary relative spatial relation between GIS and Time-of-Flight (ToF) range camera further complicated by a markerless configuration. We propose to estimate the camera’s pose solely based on matching of GIS objects and their detected location in image sequences. We develop a coarse-to-fine matching strategy that is able to match point clouds without any initial parameters. Experiments with a state-of-the-art ToF point cloud show that our proposed method delivers an absolute camera position with decimeter accuracy, which is sufficient for many real-world applications (e.g., collision avoidance).

Keywords: Indoor positioning; ToF cameras; range imaging; CityGML; point cloud library

1. Introduction

Even though indoor positioning is a comparatively new research topic, the development of indoor positioning techniques has become a major research field. Three-dimensional (3D) geometries of objects or scenes need to be captured for a variety of applications, such as scene mapping, robot navigation, and video surveillance where physically deployed infrastructure should not be required during data acquisition to minimize the costs. Several hundred approaches have been made in the past few years, as no localization technology is able to cover the indoor space like Global Navigation Satellite Systems (GNSS) do for the open sky. Mautz (2012) summarizes the state-of-the-art in indoor positioning techniques. The majority of systems relies on active transmission of electromagnetic or sound waves and often approximate methods (proximity, scene analysis, etc.) are applied to obtain a rough estimate of the unknown location (Blankenbach and Nordine, 2010). Beacon-based positioning techniques require knowledge of the geospatial location of their transmitters, which can be cumbersome to achieve. Laser scanners, measuring each point sequentially, triangulation methods (e.g., stereo-vision and photogrammetry), and interferometry are commonly used for optical based indoor positioning. Drawbacks of these techniques include time-consuming data acquisition due to the sequential scanning process of terrestrial laser scanners, challenging stereo image analysis for stereo camera systems or visual odometry (Scaramuzza and Fraundorfer, 2011), and limited depth range for interferometric methods (Kavli et al., 2008). Monocular vision systems based on smartphone camera images and their discovery in...
an image database (Werner et al., 2011; Mautz, 2012) or floor plans (Blankenbach and Norrdine, 2010; Huang and Gao, 2012) are difficult to interpret due to scale ambiguities. Additional information about landmarks (door frames, etc.) is needed and the expected accuracy of such technique is at meter level.

An alternative technique that is able to rapidly acquire large amounts of indoor depth data in a video-like fashion is range imaging (RIM). ToF range cameras measure depth information directly without need for stereo matching. Depth ranges of several tens of meters with centimeter to decimeter precision of state-of-the-art systems are largely sufficient for indoor applications.

We propose to estimate camera positions via matching of range image sequences to already available GIS data. Since relative orientations of objects and absolute position are known therein, we can use this information to pose our measuring device once newly acquired point clouds are accurately matched to the model. Such models have become widely available because various disciplines like Computer Aided Architectural Design (CAAD)/BIM, Computer Graphics, and Geographic Information Systems (GIS), which deal with 3D interior building models (e.g., IFC (buildingSMART, 2008; Scaramuzza and Fraundorfer, 2011) or CityGML (Kavli et al., 2008; Gröger et al., 2008)). Note that such models do not only store 3D shape and position of objects, but represent their precise interior topography including semantics, too. For example, a cupboard in an office does not only appear as a cuboid but is explicitly tagged as a cupboard. Thus, single objects of interest for matching to newly acquired point clouds can rapidly be found in extensive datasets and can also be processed successfully.

Our method combines absolute and relative orientation to achieve decimeter accuracy of a mono-camera system (ToF range camera) while no dedicated markers nor any other locally deployed infrastructure (e.g., Wi-Fi hot spots) inside the building is required (Guðmundsson et al., 2007). Moreover, no additional devices like inertial measurement units (IMU) or odometers are used. Matching and estimation of the camera position solely relies on range measurements and an a priori given building model. Note that a value adding service of our approach is the generation of a 3D building model from the observed point cloud.

In the following we first review works related to ours. After a conceptual overview of our approach, we provide a detailed methodological step-by-step explanation of the proposed point cloud matching and camera positioning procedure. Thereafter, experiments with a challenging dataset are described and discussed. Finally, we give conclusions and an outlook.

2. Related Work

Arman and Aggarwal (1993) proposed a definition for the exact estimation of location and orientation of an object, namely its pose, as the object recognition problem. Some prior knowledge about the object (e.g., shape, color, etc.) is relevant and can be contained in an object model. This model represents the object adequately if it is unique, not sensitive, unambiguous and convenient to use. Bosch and Haas (2008) presented a model-based approach, which automatically registers 3D CAD models with laser scanner data. Therefore, the proprietary data format from CAD is converted into the Stereo-Lithography (STL) format and is then referenced in the laser scanner’s spherical frame. The conversion to STL reduces computational complexity. Furthermore, the vertices of STL triangles are expressed with spherical coordinates.
Prusak et al. (2008) mounted a ToF range camera and a Fisheye camera to a mobile robot. The pose is estimated using a model-tracking/structure from motion (SfM) algorithm. In a first step range and fisheye images are mapped to a 3D-panorama to generate 2D-3D-correspondences therein. Further on, the 3D-points and estimated poses are used as input data for mapping the building based on a SLAM algorithm. Fuchs and May (2008) reconstructed a cube’s surface with an Iterative Closest Point (ICP) algorithm merging ToF range camera point clouds. The pose of the cameras was known a priori in a global coordinate system with a precision of 1 mm in translation and 0.1° in rotation. After depth and photogrammetric calibration of the cameras the cube was reconstructed during an ICP algorithm with an accuracy of approximately 3 mm in translation and 3° in rotation. Sheh et al. (2006) merged a generated point cloud with color and thermal image data. The ICP algorithm was used to generate textured and accurate maps of unstructured indoor environments. Therefore the data acquisition involved rotation by a pan-tilt unit, which took ten range images at intervals of 36°, stopping at each location long enough to avoid motion blur. However, a human operator, who identified landmarks, assisted the mapping procedure. Due to the drawback of ICP to often converge to an incorrect local minimum if the input point clouds are not already nearly aligned, May et al. (2009) investigated modifications in a SLAM algorithm. They provide a performance benchmark comparing Kanade-Lucas-Tomasi feature tracker (KLT) and Scale-invariant feature transform (SIFT) algorithm to a depth-image based ICP algorithm and the hybrid Efficient Second Order Minimization (ESM) technique.

In Kohoutek et al. (2010) we have shown how to construct the object database in CityGML. CityGML is a standardized information model which considers the objects’ geometry as well as their semantics, topology, and appearance (Nagel et al., 2009). In particular for the purpose of indoor modeling, the semantic model provides an object class ‘Room’ that contains attributes to classify rooms and their function, for example, as a living room or office. Objects’ (e.g., installations, furniture) geometric relation/constellation and label (name) identify a specific room or at least minimize the possible number of rooms in a building in which the camera is located.

Based on this background we will present in this work how the transformation from acquired point clouds to an object model is realized. The main challenge of our approach is that we face the problem of datasets with totally different amounts of points. Furthermore, a 3D model needs to be matched to a 2.5D scan. In the present system only geometric information is used. The advantage of CityGML, its semantic data, can be used in future for room identification.

3. Autonomous Indoor Positioning

The basic concept is to position a ToF range camera indoors via matching of range image sequences to GIS building models (Fig. 1). A coarse-to-fine matching procedure consisting of three steps is developed and will be explained in the following. Range point clouds are transformed to the GIS model, where all relative and absolute point positions are known a priori, via a 3D transformation within an iterative matching framework. Once matching is accomplished the camera pose is estimated.
The database needs to be given a priori by a Building Information Model (BIM). Such models are nowadays established during the construction phase of a building and measured by other sensors, e.g., laser scanners. Recall that we neither assume any particular markers nor other locally deployed infrastructure inside the building. Although this assumption makes our method applicable to a wide range of objects and tasks, it makes sensor orientation and positioning challenging. Furthermore, the ToF range camera is used as a self-contained sensor where no additional information from other sensors as IMUs or odometers is used. Positioning solely relies on the range camera and a given building model.

3.1. Matching

A matching procedure for different spatial datasets being somehow located apart from each other always consists of two main parts: First, a suitable transformation that is capable of mapping the input to the reference and second, a metric that measures the fitting quality between both datasets after each transformation. Such metric can also be interpreted as a score function where an optimal score is aimed at. The exterior camera orientation is determined by a Cartesian 3D coordinate transformation with three shift and three rotational parameters. Transformation parameters are improved iteratively, for example using gradient descent methods, and after each run the metric measures the quality of fit. The goal is to find those transformation parameters that lead to an optimal fit between the datasets with respect to the metric. In practice, this sequence of transformation and metric evaluation is repeated iteratively until convergence of the metric value.

It should be noted that coordinates of the building model are usually given in an absolute national coordinate system (e.g., LV95 in Switzerland (swisstopo, 2010)) whereas camera range measurements are recorded in a local camera-specific reference system. It goes without saying that we initially have to convert both datasets, the acquired 3D object point clouds from the ToF range camera $m_i$ and the a priori known object models $d_i | i = 1 ... N$, to the same reference system otherwise the following matching algorithms would not be applicable due to huge global coordinate offsets.

In a first step the absolute coordinates of the GIS object model are reduced such that the camera is located inside the building of interest. Therefore, the integer part of the first point of the object model is subtracted from all other model points thus accounting for large offsets. This translation $T^O_i$ provides an initial guess for the
translation vector $T_i$. The entire following matching process then has the goal of estimating the camera's pose with respect to the GIS object model, i.e., we match the ToF point clouds to the GIS object model. Once the global offset has been accounted for, we assume that input point cloud and target point cloud may have arbitrary orientation and position, both, relative to each other and in absolute coordinates. Literally speaking, we suppose that it is already known in which particular building the camera is located, but we do not know where exactly, e.g., in what room, on which floor etc. The transformation is calculated with:

$$d_i = R_i m_i + T_i + V_i$$  \((1)\)

where $R_i \in \mathbb{R}^{3 \times 3}$ is a standard $3 \times 3$ rotation matrix, $T_i$ is a 3D translation vector and $V_i$ a noise vector (Büttgen et al., 2005). Due to the fact that acquired point clouds and object models are metrical, a scaling operator is not needed. ToF cameras are able to measure the absolute distance (Kohoutek et al., 2010). Suitable reference points for the transformation (with six degrees of freedom) are the corners and walls of the room, vertices of doors, windows and other fixed installations or objects (e.g., furniture).

Generally, it is essential to keep prior assumptions as relaxed as possible because they could potentially limit applicability in practice. In a real-world scenario any kind of orientation and positioning from one point cloud with respect to the other one inside a building is possible and our method has to cope with this situation. Therefore, we use a coarse-to-fine matching procedure. A first coarse matching is done without need for precise initial values.

After the initial translation $T_i^0$ which basically shifts the ToF camera into the building of interest, the remaining displacement needs to be found. Most state-of-the-art algorithms establish correspondences between primitives of both datasets. A common solution for the registration problem is the Iterative Closest Point (ICP) algorithm or one of its variants (Besl and McKay, 1992). ICP iteratively refines the relative pose of two pre-aligned point clouds by minimizing the sum of squared distances of corresponding points. Corresponding point pairs are identified via Euclidean distances of neighboring points in both scans. However, if point clouds are not pre-aligned with a certain precision, ICP tends to converge in a local minimum because nearest neighbor points do not correspond to the same points in the second point cloud if datasets are located far apart. In our case there is no pre-alignment of both point clouds because of the absence of any precise initial transformation parameters. Therefore, we cannot use ICP for matching.

A point cloud matching method without need for relatively precise initial transformation parameters is proposed in Biber and Straßer (2003). The two main advantages of the so-called Normal-Distributions Transform (NDT) are neither need for pre-alignment nor establishment of point correspondences between datasets for their registration. While ICP does a point-to-point matching, NDT is based on finding linear combinations of normal distributions. Normal distributions represent a piecewise smoothing of the scan and standard numerical optimization methods can be used for registration. Furthermore, computation time is increased because there is no need for nearest-neighbor search, which is a computational bottleneck of ICP. Magnusson (2009) extends the NDT registration to 3D and introduces some refinements. Such extended NDT algorithm uses a voxel data structure to represent local surfaces of objects compactly and carries out a More-Thuente line search (Moré
The voxel grid data structure does not use individual points, but instead measures distribution statistics contained in each of its voxel cells to model point clouds as a set of multivariate Gaussian distributions. As a consequence, point cloud filtering as pre-processing step before registration is thus not necessary. NDT represents the probability of measuring a sample for each position. It is possible to adjust and optimize the probability of the existence of points at any position within the voxel. If the voxel size is chosen too small the registration will succeed only if point clouds are close to each other, similar to the pre-alignment for ICP. On the other hand it should not be chosen too big so that small objects can still be detected (Okorn, 2012). The lower cell size bound is given by the prerequisite to reliably compute a covariance matrix calling for at least five points per cell. A threshold (the epsilon parameter) is setup that defines the minimal change of the final transformation vector \( T_{1}^{\text{NDT}} \) and \( R_{1}^{\text{NDT}} \) (Okorn, 2012). The iterative alignment terminates when the incremental change reaches the threshold. The step length should shrink as it approaches the optimal solution. Larger distances can be covered by a smaller number of iterations using a larger maximum step length but at the risk of overshooting and ending up in an undesirable local minimum. We use the extended and refined NDT for coarsely registering ToF point cloud and GIS object model. It should be noted that the GIS object model originally only has points at plane intersections, which leads to several magnitudes less points than the ToF point cloud. Direct registration with such high point density difference is usually impossible and thus the GIS object model is augmented by randomly distributing points on each object plane.

Due to the fact that the point clouds have total different amounts of points, the NDT algorithm might not converge in its best solution. Therefore, NDT is used to achieve a coarse registration which is followed by fine registration with Correspondence Grouping (CG) (Cavallari and Tombari, 2011). The CG algorithm is capable to handle cases where the number of matched point correspondences is quite small compared to the total number of points like in our case. The object of interest is only a small part of the acquired ToF point cloud because the field of view of the sensor captures the entire environment like presented in Fig. 2.

Due to the fact that the ToF point cloud contains not only points of the object of interest and the CG algorithm is not based on a voxel data structure like NDT the ToF point cloud needs to be filtered. A promising algorithm is given by plane model segmentation to delete for example the floor and walls. Okorn (2012) uses two clustering algorithms, one is based on a 3D Hough voting scheme (Tombari and Di Stefano, 2010), the other one is based on evaluating the consistency of the

---

**Fig. 2 Acquired point cloud from ToF range camera.**
geometry (Chen and Bhanu, 2007). We decided to focus on 3D Hough voting to detect free-form objects in range images because of the promising results of Tombari and Di Stefano (2010). In the Hough voting approach random feature points and their local neighborhoods are extracted in GIS object model and ToF point cloud. By using a threshold, e.g., the Euclidean distance between points in a neighborhood, a set of correspondences can be determined, which is robust to wrong correspondences caused by noise or occlusions. Note that Hough voting in 3D space basically detects planes in the point clouds and results in matching of straight edges. The final transformation matrix includes values for the six degrees of freedom ($T_i^{CG}$ (x, y, z) and $R_i^{CG}$ (roll, pitch and yaw)) as needed to transform the GIS object model to the local coordinate system of the ToF sensor. The final parameter set is passed to camera pose estimation.

3.2. Estimation of Camera Position

After the first translation of the object model coordinate system into the camera coordinate system the acquired point cloud will from now on be transformed into the object model. Once point clouds have been matched the camera pose can be estimated in a user specified interval (e.g., every 80 frames) by adding the translation vector $T_i^G$. The final transformation is calculated with:

$$
\begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix}^{\text{Trans}} = \begin{pmatrix}
T_x \\
T_y \\
T_z
\end{pmatrix} + \begin{pmatrix}
1 & R_z & -R_y \\
-R_z & 1 & R_x \\
R_y & -R_x & 1
\end{pmatrix} \begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix}^0
$$

with $T_i = T_i^{CG} + T_i^{NDT} + T_i^{CG}$ and $R_i = R_i^{NDT} + R_i^{CG}$.

It has to be mentioned that the camera coordinate system of the MESA® ToF range camera is by construction not the same as usually used in the literature on perspective camera projection models. It lies in the center of the lens and X and Y are rotated around Z positively about $\pi/2$.

It is a “right-handed” camera coordinate system, with X-coordinate increasing horizontally to the left, Y-coordinate increasing vertically upwards and Z-coordinate increasing along the optical axis away from the camera. Fig. 3 displays the origin of the coordinate system (0, 0, 0) which is located at the intersection of the optical axis with the front face of the camera.

![Fig. 3 Origin (x,y,z) as delivered by the camera (Mesa® Imaging AG, 2011a).](image)

4. Experiments

We implement all previously described algorithms in the framework of the open source Point Cloud Library (PCL) containing a wide range of state-of-the-art algorithms like filtering, registration, model fitting, etc. (Rusu and Cousins, 2011). It offers well-
elaborated algorithms in C++ and is also capable of handling 3D point clouds in real time.

We evaluate the proposed method on point clouds acquired with a MESA® ToF camera SwissRanger 4000 (Mesa® Imaging AG, 2011a). Acquired point clouds have an approximate 3D position accuracy of 1 cm for distances of up to 5 m and < 1 dm accuracy for distances up to 15 m (in terms of a 1-σ standard deviation). For many indoor applications this level of accuracy is sufficient, e.g., collision avoidance. We chose a frame rate of 30 frames per second to acquire a point cloud over all 25344 pixels of the sensor.

Our chosen test object is a block of wood that is uniquely identifiable in its orientation from all view directions. The object model was generated as a small VRML model in the Swiss coordinate system LV95. The VRML model is only represented by the ten edge points and the information which point is a member of which plane (Fig. 4). However, the matching algorithm performs well with all other objects like installations and furniture. Nevertheless, the amount of points has to be increased to perform the matching algorithm successful because our approach calls for two point clouds with similar point densities as input. Therefore, up to 1000 random points will be added to each plane of the object.

![Fig. 4 VRML model of the test object with seven surfaces and ten object points (left), as wire frame model (middle) and with added random points (right).](image)

### 4.1. Sensor Specifications

The measuring principle of the MESA® ToF camera SwissRanger 4000, schematically shown in Fig. 5, is based on the phase shift between light emitted from a light source and the reflected light received at a sensor using Complementary Metal Oxide Semiconductor technology (CMOS/CCD) (R. Lange and Seitz, 2001). The emitted light is pulsed at the modulation frequency $f_{mod}$. The sensor samples the reflected light regularly and calculates the phase shift $\varphi$ of the modulation with an autocorrelation function (Möller et al., 2005). Since $\varphi$ is proportional to the target range, it is possible to calculate an absolute target distance:

$$D = \frac{c \varphi}{4 \pi}$$

where $c$ is the speed of light. In addition to the signal phase shift, the amplitude and the offset can be measured. Here, the amplitude indicates the strength of the modulated signal, which is an indication for the measurement accuracy. While the offset represents the local brightness of the scene, i.e., a gray scale value similar to gray scale images.
The maximal non-ambiguity distance $D_{\text{max}}$ of 10 m is limited to half the modulation wavelength $\lambda_{\text{mod}}$. Distances larger than $D_{\text{max}}$ are folded back to the non-ambiguity distance. Camera specifications of the device we use are listed in Table 1.

Table 1. SR4000 specifications (modified from Mesa Imaging AG (2011a)).

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modulation frequency (MHz)</td>
<td>14.5–31</td>
</tr>
<tr>
<td>Measurement range (m)</td>
<td>calibrated 0.8–8</td>
</tr>
<tr>
<td>Sensor pixels</td>
<td>176 x 148</td>
</tr>
<tr>
<td>Field of view (degree)</td>
<td>43.6 x 34.6</td>
</tr>
<tr>
<td>Scan resolution at 3 m (mm)</td>
<td>13.6</td>
</tr>
<tr>
<td>Footprint area at 3m (m²)</td>
<td>4.48</td>
</tr>
<tr>
<td>Camera weight (g)</td>
<td>470</td>
</tr>
<tr>
<td>Camera dimensions (mm)</td>
<td>65 x 65 x 68</td>
</tr>
<tr>
<td>Frame rate (f/s)</td>
<td>54</td>
</tr>
<tr>
<td>Illumination wavelength (nm)</td>
<td>850</td>
</tr>
<tr>
<td>Price (€)</td>
<td>~ 5500</td>
</tr>
</tbody>
</table>

4.2. Camera Calibration

To increase the precision of the result the cameras interior orientation has to be determined previously. The SR4000 camera used had been calibrated by the manufacturer MESA® Imaging. Fig. 6 shows the final test result from the manufacturer during an ambient temperature of 25°C (tests 9 and 10 performed at slightly higher housing temperature) without presence of background light.

Fig. 6 Final test results after calibration by manufacturer. All deviations are within the 20 mm tolerance. The absolute error represents the deviation between a reference distance and the distance measurements (Mesa® Imaging AG, 2011b).
The camera was given a warm up phase for at least one hour prior to data acquisition to ensure it reached internal temperature stability (Kahlmann and Ingensand, 2005). However, to reduce the signal to noise ratio (SNR) a mean point cloud was averaged over 100 measurements. The object was placed on the floor (dark brown colored carpet) and the ToF range camera was facing the object in a distance of 1.60 m from above (angle of incidence ca. 45°).

4.3. GIS Data Format

The a priori known GIS object models are stored as Virtual Reality Modeling Language (VRML) files with spatio-semantic information in CityGML (Gröger et al., 2008) that supports any coordinate system and also provides the missing link between the indoor and outdoor space. VRML files represent the 3D geometry of objects in simple text files to keep the data storage small. The accuracy of the objects in CityGML is expected to be at centimeter level and should lead to position determination of the camera within centimeter accuracy. In CityGML geometries of variable components of a room can be modeled using so-called implicit geometries. Only a single instance of the object’s shape is stored in the library in form of a VRML file even if multiple objects of the same shape are present (e.g., pieces of furniture). For each occurrence of such an object, only the local coordinates of an insertion point and a transformation matrix need to be stored in CityGML (Kohoutek et al., 2010). VRML was chosen instead of Extensible 3D (X3D) due to a smaller file size, which allows quick downloads of the object models via mobile Internet access. However, our approach is not restricted to VRML, objects could be modeled in X3D, too. GML database objects can be expressed in any coordinate system, in our case Swiss coordinates LV95 (e.g., X = 680589.100 m, Y = 251368.100 m, Z = 524.100 m). The 3D Cartesian coordinate system of the acquired point cloud by the SR4000 is in metrical values too, but with a maximum value of around 10 m in indoor environments (e.g., X = 1.23 m, Y = 3.67 m, Z = 7.46 m).

4.4. Matching and Positioning Results

As we mentioned before, we chose a wooden block with plane surfaces as test object for our experiments. However, our proposed matching procedure works with any kind of object which is available in the database like cupboards, chairs, tables etc. Recall that our two-step matching procedure consists of an initial coarse registration applying the Normal-Distributions Transform (NDT) (Biber and Straßer, 2003; Magnussson, 2009) followed by a fine registration via Correspondence Grouping (CG) (Cavallari and Tombari, 2011).

In order to successfully register the point cloud acquired by the ToF camera with the GIS object model point cloud, the amount of points representing the model had to be increased to achieve roughly equal point densities in both datasets. The amount of additional model points was increased by randomly distributing between 300 and 1000 points on each plane.

The modification of the scale dependent parameters of the NDT algorithm in our approach is based on the example in Okorn (2012). Fig. 7 shows a possible example of the point clouds after translation $T_i^G$. In the following transformation steps the ToF point cloud is transformed to the object model.
We performed a grid search to find the optimal NDT parameter setting, which are shown in Table 2.

**Table 2. Parameters of NDT used to produce Fig. 8.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>pcl::NormalDistributionsTransform</td>
<td></td>
</tr>
<tr>
<td>ndt.setTransformationEpsilon</td>
<td>0.01</td>
</tr>
<tr>
<td>ndt.setStepSize</td>
<td>0.3</td>
</tr>
<tr>
<td>ndt.setResolution</td>
<td>2.0</td>
</tr>
<tr>
<td>ndt.setMaximumIterations</td>
<td>100</td>
</tr>
</tbody>
</table>

Based on the chosen parameters, the algorithm took eight seconds to determine the location and calculate the transformation parameters $T_{1}^{NDT}$ and $R_{1}^{NDT}$. Fig. 8 shows the acquired point cloud (white) transformed into the model point cloud (green). It remains a visible deviation in rotation. However, the NDT algorithm provides a good approximate solution that serves as input to a refinement with the CG algorithm.

The implemented CG algorithm is based on the tutorial of Cavallari and Tombari (2011) that explains how 3D object recognition can be carried out using a PCL_Recognition module. In this work the algorithm based on 3D Hough voting scheme was used and the chosen parameters are presented in Table 3.

**Table 3. Code example for the used CG algorithm.**

```cpp
// Compute Descriptor for keypoints
pcl::SHOTColorEstimationOMP<PointType, NormalType, DescriptorType> descr_est;
dscr_est.setRadiusSearch (0.1f); // unit: (m)
// Clustering
pcl::Hough3DGrouping<PointType, PointType, RFType, RFType> clusterer;
clusterer.setHoughBinSize (0.07f);
clusterer.setHoughThreshold (16.5);
```

Recall that our ToF point cloud covers much more than only the object itself. Thus we have to filter out non-relevant parts prior to fine registration. We adopt the plane model segmentation algorithm of Rusu (2012), which is based on the number of points
per surface with equal normal directions. Once surfaces have been segmented, all large ones with more points than a certain threshold alpha are assumed to belong to wall, floor or ceiling. Surfaces with fewer points are considered belonging to our object of interest located inside the room. A suitable threshold alpha is found empirically through multiple tests. In our case we achieve optimal results with alpha = 1600. All surfaces with more than 1600 points are discarded. Fig. 9 shows the result of the CG algorithm after filtering the input point cloud and using the solution from NDT algorithm as input. The CG algorithm converges in roughly ten seconds.

Fig. 9 Solution of CG algorithm from different viewing angles; filtered SR4000 point cloud (white) and GIS object model point cloud (green).

The model point cloud fits significantly better to the SR4000 point cloud than the original output from the NDT algorithm does (Fig. 7). Due to the fact, that there are no identical points in the model object and acquired point cloud a quality for point correspondences cannot be given. Depending on the input data, other, more unique information like cluster of points and/or empty spaces and features give a better estimation for the matching quality than points. However, like in NDT algorithm a small but visible deviation in rotation remains after the transformation ($T_{1}^{CG}$ and $R_{1}^{CG}$). This can be explained through data acquisition and multipath reflections. A large portion of the infrared light is reflected first on the floor and then on the measured object into the camera. The region of the wall in Fig. 10 will be seen further away than it is in reality and turns out in a concave structure (see orange line).

Fig. 10 Multiple path reflections on a concave scene-in that case a corner between two walls. (Mesa® Imaging AG, 2011a).

Maximum overestimation is in the region where multiple reflection paths are both, maximum in number and in intensity. This explains the shape of the measured values, as shown in orange on Fig. 10 (right hand side). Due to the fact that the light travels the direct and the indirect path the apparent distance is then a weighted average of the paths, weighted by the signal strengths. This results in over-estimated distance measurements.
The Template Alignment algorithm (TA) presented in Dixon (2011) was additionally tested to determine the object’s position and orientation. TA uses a 3D template (the object model) as input and aligns it to the target cloud by applying the Sample Consensus Initial Alignment (SAC-IA) method to align source to target. The sample consensus method samples large numbers of correspondences and ranks them very quickly (Rusu et al., 2009). It maintains the geometric relations of the correspondences without testing all combinations. The maximum correspondence distance is specified as the squared distance with a value of $2\text{ cm}^2$. After calling a SAC-IA’s accessory method the final transformation matrix ($T_{\text{TA}}$ and $R_{\text{TA}}$) and fitness score are obtained. Fitness scores indicate the matching quality. It can readily be used as evaluation criterion where smaller values indicate better matching results (Dixon, 2011).

The template alignment algorithm takes the longest computation time compared with the other two tested algorithms and did not work with our input data; the ToF point cloud was transformed to a completely wrong location. We conclude that this is a result of the large difference in the density of points between the two input point clouds causing the failure of the TA approach. In order to provide a better initial solution for the algorithm the input model point cloud was exchanged with the output point cloud from the NDT algorithm, like it was done in the GC algorithm. Nevertheless, there was no improvement.

5. Conclusions

Efficient absolute positioning of a ToF range camera based on object acquisition in form of measured points in 3D Cartesian coordinates is possible. The absolute position of the camera can be calculated with decimeter accuracy based on the transformation parameters obtained in the presented coarse and fine registration with NDT and CG algorithm. The position of the ToF camera can be transformed into the reference coordinate system, i.e., the coordinate system of the spatio-semantic 3D model. The possibility of using original VRML text format, which allows data compression for the purpose of quick download from the Internet and keeping the database small-sized.

However, ToF cameras still suffer from a set of error sources that hamper the goal of infrastructure-free indoor positioning. State-of-the-art range imaging sensors measure distances unambiguously between 0.5–10 m at an accuracy level of centimeters. Besides the influence of the incidence angle (Karel et al., 2007) and scattering artifacts (Mure-Dubois and Hügli, 2007; Kavli et al., 2008) the distance measurement errors are also a result of differing reflectivity of objects in the scene (Gut, 2004). It can be seen in the input SR4000 point cloud, where vertical planes in reality are not vertical in the acquired point cloud. This is due to multipath reflections. Another influence is given by the reflective properties of the model object like material, color and gloss. Furthermore, we did no self-calibration of the camera.

Nevertheless, the algorithms provided in the point cloud library can deal with any kind of point clouds. The generated point clouds from ToF cameras are small in their data amount in comparison to laser scanner data and can therefore be processed in near real time. Due to fast processing of 3D data at high frame rates, ToF cameras are well suited for kinematic applications, such as 3D obstacle avoidance, gesture recognition (Open Geospatial Consortium Inc., 2012b) or generating indoor maps (Sheh et al., 2006; May et al., 2009).
6. Outlook

The proposed location approach using only a range imaging sensor could be improved by self-calibration as well as considering additional sensors. Adding observations of an electronic compass and/or a tilt sensor would provide approximate values of some transformation parameters and therefore stabilize the search for the correct transformation set. Furthermore, if our approach will be used for an UVS equipped with a ToF camera additional problems like so-called mixed pixels or motion artifacts have to be solved by using filtering methods (e.g., Lindner and Kolb (2009) or Kohoutek, Dröschel, et al. (2013)). Furthermore, the usage of spatial and semantic information of CityGML can be expended for pose estimation. The spatial relation of objects can help to estimate the pose more robustly, for example, if another object occludes one object but its surroundings can be detected. In such case the semantic information could support room identification, for example, if the amount of objects (e.g., tables, monitors, etc.) inside a room is determined and compared with the database. The current demonstrator implementation determines the camera pose, (position and rotation) offline on a consumer PC. The future goal is to run the location estimation on a mobile device with data link to the ToF range camera and to the object database.
Publication 5

Indoor positioning and navigation using Time-of-Flight cameras

Tobias K. Kohoutek, David Dröschel, Rainer Mautz, Sven Behnke

TOF Range-Imaging Cameras, Chapter 8, Eds. F. Remondino, D. Stoppa, Springer: Berlin/Heidelberg, Germany, 2013, DOI: 10.1007/978-3-642-27523-4_8

(Author version; for typeset version please refer to the original book chapter)

1. Introduction

The development of indoor positioning techniques is booming. There is a significant demand for systems that have the capability to determine the 3D location of objects in indoor environments for automation, warehousing and logistics. Tracking of people in indoor environments has become vital during firefighting operations, in hospitals and in homes for vulnerable people and particularly for vision impaired or elderly people (Kohoutek et al., 2010). Along with the implementation of innovative methods to increase the capabilities in indoor positioning, the number of application areas is growing significantly. The search for alternative indoor positioning methods is driven by the poor performance of Global Navigation Satellite Systems (GNSS) within buildings. Geodetic methods such as total stations or rotational lasers can reach millimeter level of accuracy, but are not economical for most applications. In recent years, network based methods, which obtain range or time of flight measurements between network nodes have become a significant alternative for applications at decimeter level accuracy. The measured distances can be used to determine the 3D position of a device by spatial resection or multilateration. Wireless devices enjoy widespread use in numerous diverse applications including sensor networks, which can consist of countless embedded devices, equipped with sensing capabilities, deployed in all environments and organizing themselves in an ad-hoc fashion (Mautz and Ochieng, 2007). However, knowing the correct positions of network nodes and their deployment is an essential precondition. There are a large number of alternative positioning technologies (Fig. 1) that cannot be detailed within the scope of this paper. An exhaustive overview of current indoor position technology is given in Mautz (2012). Further focus will be on optical methods.
Optical indoor positioning systems can be categorized into static sensors that locate moving objects in the images and ego-motion systems whose main purpose is the position determination of a mobile sensor (i.e. the camera) (Mautz and Tilch, 2011). Some optical system architectures do not require the deployment of any physical reference infrastructure inside buildings, which can be a requirement for a widespread implementation.

This article investigates the use of Time-of-Flight (ToF) cameras for ego-motion determination in indoor environments. ToF cameras are suitable sensors for simultaneous localization and mapping (SLAM), e.g. onboard of autonomous Unmanned Vehicle Systems (UVS), or the detection and localization of objects in indoor environments. They are an attractive type of sensor for indoor mapping applications owing to their high acquisition rate collecting three-dimensional (3D) data. ToF cameras consist of compact, solid-state sensors that provide depth and reflectance measurements at high frame rates of up to 50 Hz independent from surrounding light.

The approximate 3D position accuracy for objects seen by the used MESA® Imaging ToF camera SwissRanger™ SR4000 (in terms of a 1-σ standard deviation) is 1 cm for distances of up to 5 m and 1 dm for distances up to 15 m. Such a level of accuracy is sufficient for some indoor applications, e.g. collision avoidance. Currently, ranges larger than 15 m and accuracies better than 1 cm are not applicable to ToF cameras. In these cases 3D laser scanners or stereo/multiple camera systems need to be used instead. As a drawback of two-dimensional (2D) cameras, the prerequisite for multiple views induces a high computational load since point correspondences between at least two images from different perspectives have to be determined. In addition, distances to structureless surfaces cannot be measured, because the correspondence problem (Julesz, 1960) cannot be solved. Furthermore, passive 2D vision suffers from shadowing effects and sensitivity to changes in illumination. The use of 3D laser range finders (Kessler et al., 2011) that actively illuminate the scene can avoid these issues but needs mechanical moving parts and have high power consumption as well as a low frame rate due to sequential point acquisition.
Our procedure is as follows. Image features, e.g. edges, corners or flat surfaces are detected based on reflectance data for object recognition in the indoor environment. In Section 2 we will show how the indoor positioning with the ToF camera can be realized. As a novelty, the proposed method combines absolute and relative orientation of a ToF camera without the need for dedicated markers or any other locally deployed infrastructure. This can be achieved, because in comparison to other methods RIM directly provides 3D point clouds that are compared with a spatio-semantic 3D geoinformation model offered by the City Geographic Markup Language (CityGML) that supports any coordinate system and enables the missing link between the indoor and outdoor space. As higher the level of semantic information as more accurate is the geometrical integration. The entrance door of a building for example is always connected to a walkable surface. The camera motion is estimated based on depth data and will be explained within the mapping process in Section 3. Collision avoidance becomes important if the navigation path is unknown. Section 4 will show that ToF cameras are ideally suited for that task. A welcome side effect of our approach is the generation of 3D building models from the observed point cloud.

2. Positioning Inside the Room Based on a CityGML Model

The standard CityGML (Gröger et al., 2008) defines a data model and an XML data format for 3D city and topography models. CityGML defines several Levels of Detail (LoD) with the highest LoD 4 having the capability for modeling the interior of buildings. In particular for the purpose of indoor modeling, the semantic model provides an object class ‘Room’ that can capture semantic data (Gröger et al., 2007), including attributes for the intended and current use of the room such as ‘Living Room’ or ‘Office’. An object of the class ‘Room’ can be associated with its geometry in two different ways. In one way the outer shell of a room can be defined by establishing a link to a geometric object of type Solid or MultiSurface (both types are defined by the GML 3.1.1 specification (Cox et al., 2002)). Alternatively, the outer shell can be decomposed into semantic objects of the types InteriorWallSurface, CeilingSurface and FloorSurface, which are referred to geometric objects of type MultiSurface. Openings in the outer shell of a room can be modeled with the object classes ‘Window’ and ‘Door’ that can belong to one or two InteriorWallSurfaces. This data structure can be used to express topological relationships between rooms.

The semantic object class IntBuildingInstallation can be used to model permanent fixed objects belonging to a room e.g. radiators, columns and beams. In order to model the mobile components of a room such as desks and chairs, the object class BuildingFurniture can be used. IntBuildingInstallation and BuildingFurniture provide the attribute class for semantic description of the objects (Fig. 2). The geometry of these fixed installed objects can be defined by the standard GML 3.1.1. So-called implicit geometries are used to model simplified shapes of the movable objects in a room. Hereby the shape of an object is stored only once in the library even if multiple objects of the same shape are present (e.g. pieces of furniture). The shapes could be obtained directly from the 3D CAD drawings of pieces of furniture in the manufacturer’s catalog. For each occurrence of such an object, only the local coordinates of an insertion point and the object’s orientation are stored. The orientation parameters are linked to the geometry that has become an object of CityGML.
Nowadays, Building Information Models (BIMs) are created within the planning and construction phase of a building (Nagel et al., 2009). The acquisition of BIMs for already existing buildings requires manual measurements using total stations, terrestrial laser scanners or photogrammetric techniques. Fig. 3 illustrates semantic classification of CityGML exemplified with an indoor model of a room that has been obtained by total station survey.

2.1. Room Identification Through Object Detection

Object detection is the key challenge for the correct identification of the room where the sensor is located. The detection of objects can be achieved by exploiting the amplitude image. In order to identify objects such as chairs, tables, etc., the known or “learned” primitives, features and image templates that have previously stored in the database are matched with the current image. The detected object properties such as the size, geometry or quantity of a certain object are the main criteria for the comparison with the database. This way, the unknown camera position can be limited to a small number of possible rooms in the building. The room can be identified uniquely by detecting its distinct properties, e.g. position of installations. After a successful identification additional semantic and geographic information can be extracted from the 3D geo database.
2.2 Accurate Positioning Using Distance Measurements

This step compares and transforms the in real time acquired Cartesian 3D coordinates of the objects into the reference coordinate system of the database. All room and object models in the CityGML database are saved as Virtual Reality Modeling Language (VRML) files. Suitable reference points for the transformation (with six degrees of freedom) are the corners of the room, vertices of doors, windows and other fixed installations. The accuracy of the objects in CityGML should be at centimeter level and should lead to position determination of the camera with centimeter accuracy using a least squares adjustment with a redundant number of reference points to determine the 3D camera position. One requirement for the camera is that its interior orientation has been determined previously. The exterior camera orientation (three translations and three rotations) is determined by a Cartesian 3D coordinate transformation with three shift and three rotational parameters. There is no need to estimate a scale parameter, since calibrated ToF cameras measure the absolute distance.

3. Mapping and Ego-Motion Estimation

Dense depth measurements from ToF cameras enable the generation of 3D maps of the camera’s environment. However, the accuracy of measurements in unknown scenes varies considerably, due to error effects inherent to their functional principle. Therefore, a set of preprocessing steps to discard and correct noisy and erroneous measurements need to be applied in order to achieve accuracy according to the specification.

3.1 Sensor Data Processing

First, mixed pixels at so-called jump edges are filtered out. Mixed pixels are a result of false measurements that occur when the signal from the ToF camera hits an edge of an object. Then, the signal is partially reflected at the foreground, but also at the background. Both signal parts arrive at the same CCD element. The true distance changes suddenly at the object border, but the values of the mixed pixels consist of an average between the foreground and background distance. In the point cloud, these pixels appear as single unconnected points that seem to float in the air and that do not belong to any object. This is also a common problem in terrestrial laser scanning. Jump edges are filtered by local neighborhood relations comparing the opposing angles of a point \( p_i \) and its eight neighbors \( p_{i,n} \) (May et al., 2009). From a set of 3D points \( P = \{ p_i \in \mathbb{R}^3 | i = 1 \ldots N_p \} \), jump edges are detected by comparing opposing angles \( \theta_{i,n} \) of the triangle spanned by the focal point \( f = 0 \) and its eight neighbors \( P_n = \{ p_{i,n} \in \mathbb{R}^3 | i = 1 \ldots N_p : n = 1 \ldots 8 \} \) and filtered with a threshold \( \theta_{th} \):

\[
\theta_i = \max \text{acrsin} \left( \frac{\|p_{i,n}\|}{\|p_{i,n} - p_i\|} \sin \varphi \right)
\]

\[
J = \{ p_i | \theta_i > \theta_{th} \}
\]

where \( \varphi \) is the apex angle between two neighboring pixels. Since the jump edge filter is sensitive to noise, a median filter is applied to the distance image beforehand. Besides mixed pixels, measurements with low amplitude are neglected since the accuracy of distance measurements is dependent on the amount of light returning to the sensor.
ToF cameras gain depth information by measuring the phase shift between emitted and reflected light, which is proportional to the object’s distance modulo the wavelength of the modulation frequency. As a consequence, a distance ambiguity arises: measurements beyond the sensor’s wavelength are wrapped back causing artifacts and spurious distance measurements. Wrapped distance measurements can be corrected by identifying a number of so-called phase jumps in the distance image, i.e., the relative wrappings between every pair of neighboring measurements. Dröschel et al. (2010) proposed attempt a probabilistic approach that detects discontinuities in the depth image to infer phase jumps using a graphical model. Every node in the graphical model is connected to adjacent image pixels and represents the probability of a phase jump between them. Belief propagation is used to detect the locations of the phase jumps, which are integrated, into the depth image by carrying out the respective projections, thereby correcting the erroneously wrapped distance measurements. The application of phase unwrapping for an indoor scene is shown in Fig. 4.

![Phase unwrapping of an indoor scene. (a) Image of the scene. (b) and (c) 3D point clouds that have been generated based on the camera’s depth image. Color of the points indicates the result of the algorithm; wrapped measurements are shown in red. Bright brightness encodes distance to the camera center. (b) Point cloud without unwrapping. Measured distances beyond the sensor’s non-ambiguity range are wrapped into it, which results in artifacts between distances of 0 and 3 meters. (c) Unwrapped depth image.](image)

3.2 Mapping and Ego-Motion Estimation

To estimate the camera’s motion between two consecutive frames, image features in the reflectance image of the ToF camera are extracted to determine point correspondences between the frames. To detect image features, the Scale Invariant Feature Transform (SIFT) (Lowe, 2004) is used. SIFT features are invariant in rotation and scale and are robust against noise and illumination changes.

In order to estimate the camera motion between two frames, the features of one frame are matched against the features of the other frame. The best match is the nearest neighbor in a 128-dimensional keypoint descriptor space. To determine the nearest neighbor, the Euclidean distance is used. In order to measure the quality of a match, a distance ratio between the nearest neighbor and the second-nearest neighbor is considered. If both are too similar, the match is rejected. Hence, only features that are unambiguous in the descriptor space are considered as matches.

Fig. 5 (a) and (b) show the reflectance image of two consecutive frames with detected features. Fig. 5 (c) shows the matching result of the two images. Each match constitutes a point correspondence between two frames. By knowing the depth of every pixel, a point correspondence in 3D is known.
The set of points from the current frame is called the data set, and the set of corresponding points in the previous frame is called the model set. The scene is translated and rotated by the sensor’s ego motion. Thus, the sensor’s ego motion can be deduced by finding the best transformation that maps the data set to the model set. A common approach for estimating a rigid transformation uses a closed form solution for estimating the $3 \times 3$ rotation matrix $R$ and the translation vector $t$, which is based on singular value decomposition (SVD) (Arun et al., 1987). The distances between corresponding points, after applying the estimated transformation are used to compute the root mean square error (RMSE), which is often used in range registration to evaluate the scene-to-model consistency. It can be seen as a measure for the quality of the match: if the RMSE is significantly high, the scene-to-model registration cannot be consistent. On the other hand, a low RMSE does not imply a consistent scene-to-model registration, since it also depends on the number and distribution of the point correspondences.

With the estimated ego motion between consecutive frames, accumulating 3D points of every frame generates a point-based map. A resulting map is shown in Fig. 6.

Fig. 5 SIFT feature extraction and matching applied on two consecutive camera frames on a ToF reflectance image. The numbers of detected features are 475 (a) and 458 (b). (c) Matching result: 245 features from image (a) are matched to features from image (b). White lines indicate feature displacement.

Fig. 6 The resulting 3D map based on the estimated trajectory (red). The colors of the points correspond to the distance of the point from the ground plane.
4. 3D Collision Avoidance

If the navigation path is unknown in dynamic environments, collision avoidance becomes important. ToF cameras are ideally suited for collision avoidance since they measure distances to surfaces at high frame rates. A typical example of a point cloud taken in an indoor environment is shown in Fig. 7 (a). This point cloud can be used to build a so-called height image as shown in Fig. 7 (b). A point \( p_{ij} \) is classified as belonging to an obstacle if

\[
(W_{\text{max}} - W_{\text{min}}) > H
\]

where \( W_{\text{max}} \) and \( W_{\text{min}} \) are the maximum and minimum height values from a local window \( W \), spanned by the 8-connected neighborhood around \( p_{ij} \). The threshold \( H \) thereby corresponds to the minimum tolerable height of an obstacle. It needs to be chosen appropriately since it should not be smaller than the sensor’s measurement accuracy. Due to evaluating a point’s local neighborhood, floor points are inherently not considered as obstacles. Points classified as belonging to obstacles are shown in Fig. 7 (c).

The resulting obstacle points are used to extract a 2D virtual scan similar to an obstacle map by 1.) projecting the 3D data into the xy-plane and 2.) extracting relevant information.

The number of range readings in the virtual scan as well as its apex angle and resolution correspond to the acquired 3D data. For the SR4000, the number of range readings is 176, which is the number of columns in the image array. The apex angle and the angular resolution are 43° and 0.23°, which correspond to the camera's horizontal apex angle and resolution. For every column of the ToF camera’s distance image, the obstacle point with the shortest Euclidean distance to the robot is chosen. This distance constitutes the range reading in the scan. If no obstacle point is detected in a column, the scan point is marked invalid.

The resulting virtual scan is fused with a 2D laser range scan obtained at 30 cm height yielding a common obstacle map modeling the closest objects in both sensors. The obstacle map from the 2D laser range finder and the ToF camera for the aforementioned example scenario is visualized in Fig. 8. By fusing the information of both sensors, the robot possesses correct information about traversable free space (light gray) in its immediate vicinity.
5. Conclusions and Outlook

Efficient and precise position determination of a ToF camera is possible based on kinematic object acquisition in form of 3D Cartesian coordinates. The absolute position of the camera can be obtained by a transformation from the camera coordinate system into the reference coordinate system, i.e. the coordinate system of the spatio-semantic 3D model. Positions of detected objects are reported in respect to the coordinate system of the 3D model. The described mapping approach can also be used for data acquisition of such 3D building models. The advantage of such models is the use of the VRML file text format allowing data compression for the purpose of quick Internet transfer and maintenance of a small-sized database. We conclude that rooms can be identified by detection of unique objects in images or point clouds. Such method is to be implemented in further research based on a data set, which includes multiple rooms.

Due to their measurement of a volume at high frame rates, ToF cameras are well suited for applications where either the sensor or the measured objects moves quickly, such as 3D obstacle avoidance or gesture recognition (Dröschel et al., 2011).

Difficulties of the proposed method arise from ToF cameras suffering from a set of error sources that hamper the goal of infrastructure-free indoor positioning. Current RIM sensors are able to measure distances unambiguously between 5 – 15 m at an accuracy level of centimeters. Until now so-called mixed pixels posed a problem in the literature. Filtering methods presented in Section 3 could solve this problem.

Acknowledgements

The support of Andreas Donaubauer, Andreas Schmidt, Dirk Holz and Stefan May is gracefully acknowledged.
5 CONCLUDING REMARKS

Synopsis of results

In this thesis three different applications using Time-of-Flight range cameras in indoor environments are presented. Time-of-Flight range cameras are well suited for kinematic applications. However, motion compensation needs to be taken into account to unambiguously measure distances between 0.5 – 10 m at an accuracy level of centimeters. The acquired point clouds are small in their amount of data compared to laser scanner data such that all experiments based on image and point cloud processing could be carried out in near real-time.

In the first application (Publication 1) the position of a robot was detected and its path could be monitored to avoid collision with objects or humans entering its working space. Feature points were tracked using Optical Flow algorithm and a security zone was adapted to the robot’s movements. The camera was mounted in a fixed position facing the robot from the front. The specific setup chosen in the experimental part of this paper was not ideal; a camera position above the robot is proposed. Objects and their positions were detected in respect to the camera’s local coordinate system.

Tracking human body parts in 3D is common nowadays in the gaming industry for Human Computer Interaction. To realize applicable pointing devices the face/eye tracking and hand gesture recognition needs to be improved in accuracy and reliability. In 2009, briefly before Microsoft’s Kinect™ was launched in the consumer market, it was suggested in Publication 2 to use Time-of-Flight range cameras as an input device for Human Computer Interaction. The user’s line of gaze is calculated by detecting the user’s fingertip of the index finger and eyes’ sight. The Time-of-Flight range camera was placed on top of the screen facing the user and the cursor could be directed by pointing towards the monitor.

As opposed to applications with a static camera, Publications 3 – 5 present algorithms and applications for determining the motion of the camera itself (ego-motion). Using such algorithms the Time-of-Flight range camera can be used for example on an Unmanned Vehicle System observing the environment for efficient and precise absolute positioning in indoor environments. Within Publication 4 spatio-semantic information was used to calculate the camera’s position by spatial resection and transformation into the absolute Cartesian coordinate reference system. Instead of using a standard Iterative Closest Point algorithm for registration of point clouds (e.g. O’Leary (2012)), Normal Distributions Transformation and Correspondence Grouping algorithm offered a more robust registration. For both algorithms input point clouds do not need a good initial estimation and therefore both algorithms will not lead to converge in an incorrect local minimum. However, the presented approach is not yet a real-time application and needs around 5 minutes to conclude. The chosen voxel size in NDT is essential for the point cloud’s surface representation and the resulting matching outcome. The matching of sharp edges in CG causes the final transformation step. In spite of measurement errors and multipath effects caused by the ToF measurement decimeter accuracy was achieved using a Building Information Model stored in the CityGML format.

Observing the unknown environment (e.g. around an Unmanned Vehicle System) is important for collision avoidance in particular when navigating an unknown path.
Mobile robots are equipped with additional sensors, like electronic compass and/or tilt sensors or an inertial navigation system (INS) to provide approximate values of the rotational transformation parameters and therefore facilitate the search for the correct transformation set.

Time-of-Flight range cameras are ideal for collision avoidance due to high frame rates in 3D data acquisition, as presented in Publication 1. In Publication 5 we used Simultaneous Localization And Mapping to generate 3D point based maps in real-time during the ego-motion estimation.

**Outlook**

ToF cameras can be used in indoor and outdoor environments (Nitsche et al., 2010; Nitsche et al., 2012) and can connect coordinate systems of both spaces. However, there are some major disadvantages of Time-of-Flight range cameras with respect to other sensors, which are used for similar applications. They are affected by a number of systematic effects, like multipath effects, changes in temperature, distortion, differing reflectivity of the observed object etc. that limit the attainable accuracy and hamper robust infrastructure-free indoor positioning. Therefore the most challenging task for future research is the provision of reliable indoor positioning capabilities. Furthermore, object classification needs to be improved for example by self-calibration of the ToF range camera and Look-up tables for different materials and colors.

Combining a ToF range camera with a normal 2D color camera can improve the theoretical 3D point accuracy. Furthermore it will enhance the directional resolution (Lipkowski and Scherer, 2012). Information on color, pattern and texture will be added to distance information and can improve object segmentation and identification (Hauke and Bömer, 2005). Izadi et al. (2011) presented with KinectFusion (KinFu) a geometrically precise real-time 3D modeling tool on Kinect’s® depth measurement. Object reconstruction grows continually with each added depth measurement while tracking the six degrees of freedom pose of the camera. RGB camera data can be used to texture the reconstructed model or indoor scene. Scanned and modeled data enable realistic forms of Augmented Reality (AR) in which real-world physics aspects can be simulated. Heredia and Favier (2012a) implemented an extension to large areas to KinFu. The new module named KinFu Large Scale offers all functionalities of KinFu to produce a mesh of a room. However, KinFu Large Scale is more time consuming and less robust to fast movements of the camera (Heredia and Favier, 2012b). Nakano et al. (2012) showed the potential of Kinect™ Positioning System (KPS), detecting multiple moving targets in a range of up to 10 m.

The wish for bigger sensor size and longer measurement range remains on the research and consumer part. State-of-the-art ToF range cameras that analyze the phase difference of a modulated continuous wave have a manufacture calibrated unambiguity range of up to 10 m and a maximum sensor size of 200 x 200 pixels. Longer measurement range (up to several hundred meter (McCarthy et al., 2009)) can be achieved with ToF range cameras using Time-Correlated Single Photon Counting (TCSPC) based on Single-Photon Avalanche Diodes (SPADs) in standard CMOS technology. But also TCSPC techniques have disadvantages like for example detector dead time where the detection system is shut down to reset after the recording of an event (Gatt et al., 2009) or limited spectral range of practical single-photon detectors (Buller et al., 2007).
The Open Geospatial Consortium will provide a suitable database and standard in communication and protocols for indoor positioning along with the development of IndoorGML. This will simplify providing suitable databases, which can be used by a variety of sensor systems for indoor positioning.
ACKNOWLEDGMENT

I would like to express my thanks to several people who influenced my work and the outcome of this thesis. Namely, these are:

• Hilmar Ingensand for giving me the opportunity to grow in his research team and his confidence towards me; for being my “Doktorvater”, and for the space he allowed to develop this thesis,

• Jan Böhm for presenting me the first idea of indoor positioning with ToF range cameras already in 2007 and his willingness to be my co-referent,

• Bertrand Merminod for his willingness to be my second co-referent, more or less out of a sudden and making it possible for me to go through the official regulations,

• All members of the IGP staff for the good time I had during the last 6 years; special thanks go to the members of the geomETH/GSEG group and the family friendly environment; in particular to Rainer Mautz for his helpful comments and continuous linguistic support throughout this thesis as well as to Andreas Wieser as new head of the group and his helpful input,

• All my friends and family who were with me during all the ebbs and flows that accompany a PhD thesis, in particular Paula for her faithful support and trust in writing this work.
ABBREVIATIONS

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Expansion / Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D</td>
<td>Two Dimensional</td>
</tr>
<tr>
<td>3D</td>
<td>Three Dimensional</td>
</tr>
<tr>
<td>AR</td>
<td>Augmented Reality</td>
</tr>
<tr>
<td>BIM</td>
<td>Building Information Model</td>
</tr>
<tr>
<td>b-rep</td>
<td>Boundary Representation</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer Aided Design</td>
</tr>
<tr>
<td>CAAD</td>
<td>Computer Aided Architectural Design</td>
</tr>
<tr>
<td>CCD</td>
<td>Charge Coupled Device</td>
</tr>
<tr>
<td>CityGML</td>
<td>City Geography Markup Language, a common information model for the representation of 3D urban objects</td>
</tr>
<tr>
<td>CLIPS</td>
<td>Camera and Laser Indoor Positioning System</td>
</tr>
<tr>
<td>CMOS</td>
<td>Complementary Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>CG</td>
<td>Correspondence Groups</td>
</tr>
<tr>
<td>CSEM</td>
<td>Centre Suisse d’Electronique et de Microtechnique SA</td>
</tr>
<tr>
<td>CSG</td>
<td>Constructive Solid Geometry</td>
</tr>
<tr>
<td>DOI</td>
<td>Digital Object Identifier</td>
</tr>
<tr>
<td>DOF</td>
<td>Degree Of Freedom</td>
</tr>
<tr>
<td>ESM</td>
<td>Efficient Second Order Minimization</td>
</tr>
<tr>
<td>FOV</td>
<td>Field Of View</td>
</tr>
<tr>
<td>GIS</td>
<td>Geographic Information System</td>
</tr>
<tr>
<td>GML</td>
<td>Geography Markup Language</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System, any of the existing or proposed satellite-based positioning systems, such as GPS, GLONAS, Galileo and Beidou</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning Service</td>
</tr>
<tr>
<td>HCI</td>
<td>Human-Computer Interaction</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>ICP</td>
<td>Iterative Closest Point</td>
</tr>
<tr>
<td>IFC</td>
<td>Industry Foundation Classes</td>
</tr>
<tr>
<td>IGP</td>
<td>Institute of Geodesy and Photogrammetry at ETH Zurich</td>
</tr>
<tr>
<td>IndoorGML</td>
<td>Indoor Geography Markup Language, schema framework for interoperability between indoor navigation applications</td>
</tr>
<tr>
<td>IMU</td>
<td>Inertial Measurement Unit</td>
</tr>
<tr>
<td>INS</td>
<td>Inertial Navigation System</td>
</tr>
<tr>
<td>KinFu</td>
<td>KinectFusion</td>
</tr>
<tr>
<td>KLT</td>
<td>Kanade-Lucas-Tomasi Feature Tracker</td>
</tr>
<tr>
<td>LBS</td>
<td>Location Based Services</td>
</tr>
<tr>
<td>LED</td>
<td>Light Emitting Diode</td>
</tr>
<tr>
<td>LoD</td>
<td>Level of Detail</td>
</tr>
<tr>
<td>LV95</td>
<td>Landesvermessung LV95</td>
</tr>
<tr>
<td>MOCCD</td>
<td>Multiocular Contracting Curve Density algorithm</td>
</tr>
<tr>
<td>MS</td>
<td>Microsoft</td>
</tr>
<tr>
<td>NDT</td>
<td>Normal-Distributions Transform</td>
</tr>
<tr>
<td>NLoS</td>
<td>Non Line of Sight</td>
</tr>
<tr>
<td>OGC</td>
<td>Open Geospatial Consortium</td>
</tr>
<tr>
<td>OpenCV</td>
<td>Open Source Computer Vision Library</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>PEM</td>
<td>Point-based Environmental Model</td>
</tr>
<tr>
<td>PCL</td>
<td>Point Cloud Library</td>
</tr>
<tr>
<td>PMD</td>
<td>Photonic Mixer Device</td>
</tr>
<tr>
<td>RCX</td>
<td>Robotic Command Explorer</td>
</tr>
<tr>
<td>RFID</td>
<td>Radio Frequency Identification</td>
</tr>
<tr>
<td>RGB-D</td>
<td>RGB color space and depth or distance information</td>
</tr>
<tr>
<td>RIM</td>
<td>Range Imaging</td>
</tr>
<tr>
<td>RIS</td>
<td>Robotic Invention System</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root Mean Square Error</td>
</tr>
<tr>
<td>RSSI</td>
<td>Received Signal Strength Indicators</td>
</tr>
<tr>
<td>SAC-IA</td>
<td>Sample Consensus - Initial Alignment</td>
</tr>
<tr>
<td>SFM</td>
<td>Structure From Motion</td>
</tr>
<tr>
<td>SIFT</td>
<td>Scale Invariant Feature Transform</td>
</tr>
<tr>
<td>SLAM</td>
<td>Simultaneous Localization And Mapping</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>SPADs</td>
<td>Single-Photon Avalanche Diodes</td>
</tr>
<tr>
<td>SR</td>
<td>SwissRanger</td>
</tr>
<tr>
<td>STL</td>
<td>Stereo-Lithography</td>
</tr>
<tr>
<td>SVD</td>
<td>Singular Value Decomposition</td>
</tr>
<tr>
<td>SWG</td>
<td>Standards Working Group</td>
</tr>
<tr>
<td>TA</td>
<td>Template Alignment</td>
</tr>
<tr>
<td>TCSPC</td>
<td>Time-Correlated Single Photon Counting</td>
</tr>
<tr>
<td>ToF</td>
<td>Time-of-Flight</td>
</tr>
<tr>
<td>USB</td>
<td>Universal Serial Bus</td>
</tr>
<tr>
<td>UVS</td>
<td>Unmanned Vehicle Systems</td>
</tr>
<tr>
<td>VBP</td>
<td>Vision Based Protective Device</td>
</tr>
<tr>
<td>VR</td>
<td>Virtual Reality</td>
</tr>
<tr>
<td>VRML</td>
<td>Virtual Reality Modeling Language</td>
</tr>
<tr>
<td>WFS</td>
<td>Web Feature Service</td>
</tr>
<tr>
<td>WGS84</td>
<td>World Geodetic System 1984</td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
</tr>
<tr>
<td>X3D</td>
<td>Extensible 3D</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible Markup Language</td>
</tr>
</tbody>
</table>
REFERENCES


Guillaume, S., 2006. Identification and Tracking of Moving People in Range Imaging Data. Student thesis. ETH Zurich, Zurich, Switzerland.


KUKA Robot Group, 2006. KUKA.SafeRobot Version 1.1 0, KUKA Robot Group, Augsburg, Germany.


Mautz, R., 2010. Camera and Laser Indoor Positioning System (CLIPS) - ETH Research Database, ETH Zurich, Zurich, Switzerland.


Mesa® Imaging AG, 2009. SR4000 Data Sheet-Version 1.3. MESA® Imaging AG, Technoparkstr. 1, 8005 Zurich, Switzerland.

Mesa® Imaging AG, 2011a. SR4000 Data Sheet-Version 5.1. MESA® Imaging AG, Technoparkstr. 1, 8005 Zurich, Switzerland.

Mesa® Imaging AG, 2011b. SR4000 Final Test Report. MESA® Imaging AG, Technoparkstr. 1, 8005 Zurich, Switzerland.


OpenCV dev team, 2011. The OpenCV Tutorials, OpenCV.


