A mobile robot based system for fully automated thermal 3D mapping☆

Dorit Borrmann[a,*], Andreas Nüchter[a], Marija Đakulović[b], Ivan Maurović[b], Ivan Petrović[b], Dinko Osmanković[c], Jasmin Velagić[c]

[a] *Robotics and Telematics group, University of Würzburg, Germany*
[b] *Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia*
[c] *Faculty of Electrical Engineering, Sarajevo, Bosnia and Herzegovina*

Abstract

It is hard to imagine living in a building without electricity and a heating or cooling system these days. Factories and data centers are equally dependent on a continuous functioning of these systems. As beneficial as this development is for our daily life, the consequences of a failure are critical. Malfunctioning power supplies or temperature regulation systems can cause the close-down of an entire factory or data center. Heat and air conditioning losses in buildings lead to a large waste of the limited energy resources and pollute the environment unnecessarily. To detect these flaws as quickly as possible and to prevent the negative consequences constant monitoring of power lines and heat sources is necessary. To this end, we propose a fully automatic system that creates 3D thermal models of indoor environments. The proposed system consists of a mobile platform that is equipped with a 3D laser scanner, an RGB camera and a thermal camera. A novel 3D exploration algorithm ensures efficient data collection that covers the entire scene. The data from all sensors collected at different positions is joined into one common reference frame using calibration and scan matching. In the post-processing step a model is built and points of interest are automatically detected. A viewer is presented that aids experts in analyzing the heat flow and localizing and identifying heat leaks. Results are shown that

☆This work was supported by the SEE-ERA.NET project ThermalMapper under the project number ERA 14/01 [45] and the DAAD/MZOS grant Thermal Perception, Mapping, and Exploration under the project ID 54367543.

*Corresponding author: borrmann@informatik.uni-wuerzburg.de

demonstrate the functionality of the system.

Keywords: 3D thermal mapping, autonomous mobile robots, calibration, next-best-view-planning, visualization

---

1. Introduction

Imagine a technology that automatically creates a full 3D thermal model of an environment and detect temperature peaks in it (cf. Figure 1). Such a system would be a big step in monitoring and inspection of existing buildings and technical assets as well as in achieving energy efficiency in building construction. For example, data centers and factories rely on correct functioning of their infrastructure. Damaged pipes and cables or other parts endanger their functionality, cause pauses in the work flow and may lead to harmful fires. In many cases leaks and overheating could be detected in their early stages by use of thermography thus preventing further damage. Building construction has undergone major changes in recent years. The importance of energy efficiency has attracted notice. To meet the Passivhaus, the Zero-energy building, or even the Energy-plus building standard modern building design makes use of all available heat sources including electrical equipment or even the body heat from people and animals inside the building. While this leads to changes in the way buildings are designed it also poses the question how existing buildings can be modified to meet these standards and to eliminate heat and air conditioning losses. According to the Action Plan for Energy Efficiency [14] of the European Commission the largest and cost-effective energy savings potential lies in residential ($\approx 27\%$) and commercial ($\approx 30\%$) buildings. The system proposed in this article is meant to aid in reaching these savings.

The current state of the art for analyzing temperature related issues is thermal imaging. Fouad and Richter present a guideline for thermography in the building industry in Germany [21]. To detect thermal bridges of exterior building walls outdoor thermography is commonly used. Thermal bridges lead to a loss of energy and can cause humidity and mold growth. Only a few images are necessary to capture the entire building but at the expenses of the resolution. To detect flaws in the construction a difference of 15 Kelvin is necessary between indoor and outdoor temperature to come to significant conclusions. It is desired that the weather conditions remain stable over a longer period of time, making the morning hours in the winter months ideal.

Keeping stable conditions is easier to achieve for indoor thermography. The analysis of back-ventilated walls and roofs is only possible from indoors. Thermal bridges at exterior walls and interior walls connecting heated and unheated rooms, pillars that interrupt the thermal insulation of a building, air leaks at windows and doors and the moisture penetration at basement walls are the common applications for indoor thermography, that focus on energy efficiency in existing buildings. Other applications aim at documenting and examining the run of heating pipes, detecting blocked pipes and construction units in a building to eliminate flaws, to make room for improvements and, in some cases, to ensure safety. For new buildings or the energetic restoration of existing buildings it has also become common to perform thermography before, during and after the construction phase for quality management.

Monitoring and analysis of the data for a large building is tedious. For indoor thermography the room should be prepared at least six hours before the inspection to achieve best results. A constant temperature is desired for this period. Furniture has to be moved away from the walls to allow their inspection. The difference between indoor and outdoor temperature has to be at least 10 °C. During the inspection each room is examined with the thermal camera. For each picture the inspector has to note the exact position and orientation from where the picture was taken [20]. After the inspection the images have to be analyzed taking into account the room temperature, the humidity, the material of the wall and the angle from which objects are seen. For some applications, e.g., inspection during construction or renovation, it is also necessary that the changes are documented over time, thus asking for comparability between independently acquired datasets [21].

Thermal images document the precise temperatures without any spatial dimensions. To identify a heat source and measure its extent it is necessary that the expert analyzes the scene on-site which is time-consuming and in cases even dangerous. For applications that require repeated thermography over time, comparison of the 2D data is only possible to some extent. Successfully modifying a building with respect to thermal issues involves extensive planning. This planning would greatly be improved by the existence of a geometrically correct thermal model. We propose a robotic system that creates a full 3D model of the environment with color and thermal information (cf. Figure 1) enabling an expert to fully analyze the recorded scene offline on a computer. Furthermore, regions of interest, e.g., regions with temperature peaks or drastic changes in temperature, are automatically detected

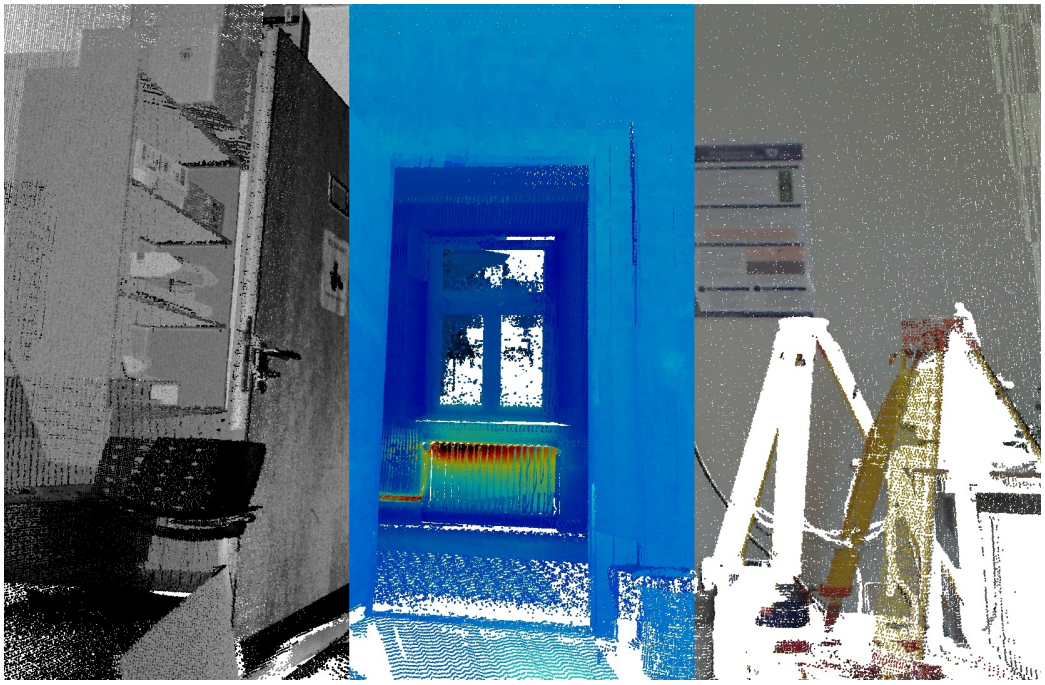


Figure 1: Laser scan with reflectance (left), thermal (middle) and color (right) information.

and pointed out to the user, increasing the efficiency of the analysis. The spatial accuracy of the model enables one to measure the extent of the area in question and its immediate localization in the building.

Our system creates a 3D thermal model of the area autonomously. The full 3D model makes it easy to identify the location of each picture as it is shown within its surroundings and its position is known. This is especially important since indoor photos capture only a small part of the scene. Automatic registration, as known from robotics, enables the merging of two models acquired at different times. According to Fouad and Richter [21] thermography can only be an auxiliary device. The analysis of the scene is only possible through expertise and experience. By automatically pointing out regions of interest in the data our system helps to find damages quickly during the analysis.

This article presents our approach to fully autonomous 3D thermal modeling of buildings. We combine thermal imaging with the technology of terrestrial laser scanning. The system has to fulfill three main tasks. First, the

data needs to be acquired and registered, i.e., all data from different sensors and positions needs to be put into one common reference frame. Second, the positions where sensor data is collected need to be planned and driven to autonomously. Third, the data has to be analyzed and presented to a user. Experiments evaluate the performance of the system and results show exemplarily a complete 3D model of an office environment collected by the system. We extend our initial work [9] by several components. The system was completed with a photo camera that allows better identification of objects in the resulting model. Errors in the calibration procedure were reduced with a novel method and the automatic detection of interesting temperature distributions was implemented in the post-processing step. The main contribution is the extension of the exploration strategy to consider the full 3D structure of the environment. Instead of exploring solely based on the floor plan of the building, the system tries to find positions from where the entire 3D space can be seen, reducing the amount of occlusions in the captured data.

2. Background and State of the Art

To access the energy efficiency of houses thermal cameras are commonly used. These cameras measure temperatures precisely, but return only 2D images of the environment and therefore the loss of energy can only be roughly quantified. Images are projections to 2D. From a sequence of images it is in principle possible to perform a 3D reconstruction. These approaches are called bundle adjustment or structure from motion (SFM). Bundle adjustment uses image features to calculate 3D positons of them. SFM adds automatic data association and therefore solves the simultaneous localization and mapping (SLAM) problem, i.e., the problem of recovering the 3D structure of the environment and the sensor poses (position and orientation). Since reliable solutions to image based 3D reconstruction from thermal images alone have not been presented yet, a second sensor has to be used to create the 3D model. Instead of using a co-calibrated system of a thermal camera and an RGB camera we use the emerging technology of terrestrial laser scanning.

2.1. Autonomous data acquisition and mapping

Laser scanning methods are well established in the surveying community and in robotics. Terrestrial 3D laser scanning systems yield precise 3D point clouds. Scanning from different poses enables one to digitize a complete

indoor environment and to resolve occlusions. Registration algorithms from the geodesy and robotics community are available to automatically align scan views from different poses.

Related work in inspection robotics includes human detection with thermal cameras using temperature signatures [31]. Högner and Stilla present a modified van as surveying vehicle for acquiring thermal images in urban environments [24]. However, the focus is on outdoor environments and image-based techniques like SFM. Prakash et al. present stereo imaging using thermal cameras, but focus on small scale applications [44]. Iwaszczuk et al. suggest an approach to map terrestrial and airborne infrared images onto existing building models [25]. The model is textured by extracting polygonal parts from the image and mapping those onto the model using standardized masked correlation. Only little work has been done to combine 3D scanners and thermal cameras. Carbelles et al. present a methodology to exhaustively record data related to a World Heritage Monument using terrestrial laser scanning, close range photogrammetry and thermal imagery [11]. They use four different sensors for data acquisition: a reflectorless total station, a terrestrial laser range scanning sensor, a digital photo camera and a thermal camera. With a total of eight natural control points a total station they relate the geometry between different sensors. Pelagottia et al. present a first automatic approach for multispectral texture mapping [43]. Their method is based on the extraction of a depth map in the form of an image from the model geometry, whose pixels establish exact correspondences with the vertices of the 3D model. The registration with the chosen texture is performed based on the maximization of mutual information.

3D environment mapping using 3D scanners on mobile robots are subject to research [51, 36]. Building thermal 3D models of environments has received some attention recently. Ham and Golparvar-Fard model and evaluate thermal models and the energy performance of buildings [23]. For this purpose they co-calibrate a thermal camera with an RGB camera. The color images are used to create a 3D model using SFM. SFM approaches are prone to failure in regions with few features or repeating structures, the density of the resulting model is low and the scale is unknown. Vidas et al. [56, 48] focus on co-calibrating a thermal camera and a Microsoft Kinect. They developed a hand-held system that creates a 3D model of the environment based on registering the Kinect data. This approach yields a dense model but is limited to the accuracy of the Kinect camera and requires a human operator. Laser scanning has the advantage that the resulting dense model

has a high geometric accuracy and is not as sensitive to repeating structures and feature-less areas as SFM approaches are. Gonzáles-Aguilera et al. [15] combine the technology of laser scanners and thermal cameras to create models of building exteriors. They extract features from both the thermal images and the projections of the point cloud from the laser scanner and match these to register the data. In indoor environments this approach is prone to errors as only small parts will be visible due to the small opening angles of thermal cameras. In combination with the low resolution of typical thermal cameras images tend to have too few features for a reliable registration. To the best of our knowledge, a fully autonomous system for modeling using 3D scanning and thermal imaging has not been done yet.

2.2. Sensor placement planning

Sensor placement planning is needed for the goal directed acquisition of 3D data. The task of a sensor placement planning algorithm is to find a set of sensor configurations needed for obtaining a detailed environment model. Since a typical 3D laser scan takes 3 to 5 minutes for one position, depending on the resolution, it is desirable to minimize the number of scanning positions. This leads to an optimization problem similar to the Art Gallery Problem (AGP) (where to place guards such that the entire gallery is guarded). The AGP problem is NP hard and is usually solved by heuristics that perform well in practice [22]. These methods are categorized as model-based sensor placement planning (*a priori* model of the environment is known) and non-model-based methods. The latter are applied for exploration tasks in which the robotic system has to navigate autonomously in an unknown environment and build its own model. The planner must determine the next-best-view (NBV) based on the information collected from previous scans. Most exploration strategies push the robot onto the border between explored and unexplored regions [18, 54]. The majority of exploration algorithms is not reliable when applied under real conditions due to the sensitivity to uncertainty of measurements, localization, and map building. A small divergence in localization at the pre-computed NBV point can lead to many unnecessary movements. Moorehead et al. include the uncertainty of the robot pose into the exploration strategies [34]. Recently, numerous sensor placement planning algorithms have been developed for the reconstruction of 3D environment models. Most methods take 3D scans based on a 2D exploration strategy [51]. For creating a full 3D thermal model of a building it is essential to consider the 3D geometry of the environment to ensure that all parts

of the building are mapped. Blaer and Allen propose a 3D NBV method which plans additional viewing locations based on a voxel-based occupancy procedure for detecting holes in the model [7]. Low and Lastra present a full non-model-based 3D NBV method based on an exhaustive hierarchical 3D view metric evaluation [29]. However, the computational complexity is still the major challenge in designing practical 3D NBV solutions.

2.3. Surface reconstruction and model building

The process of generating a surface from a set of 3D points is called surface reconstruction. The last challenging task in building a 3D thermal model of an indoor environment is the reconstruction of the 3D mesh with the temperature scalar field mapped onto it. Methods to reconstruct the 3D surface from a point cloud can be divided into two distinct groups, namely face-based reconstruction (triangulation) and iso-surface reconstruction (meshing). The most common triangulation methods use are the Voronoi [3] and 3D Delaunay [4] algorithms. Su and Drysdale present several implementations of the latter algorithm analyzing the performance [50]. These algorithms require closed and organized geometry, i.e., that the model has to be fully connected and without major outliers in the point cloud. The algorithms tend to close the structure of the model, which is good for the digitization of cultural heritage sites as presented in [13] and [2] but not sufficient for environment reconstruction in terms of time and memory efficiency as well as overall output quality. Iso-surface reconstruction, unlike the triangulation which reconstructs part-by-part, generates iso-surfaces, meaning that the model is generated by connecting the points with constant values, i.e., distance, pressure, temperature, etc.. The most popular algorithm for iso-surface reconstruction is the Marching Cubes Algorithm (MCA) developed in [28]. This algorithm starts with dividing the input space into cubes. After selecting a random cube and its eight neighbors, the algorithm determines the polygonal part of the iso-surface passing through the cube.

There are also modifications to the standard MCA presented in [12] and [35] that deal with "ripples" caused by topological inconsistencies of the model (or point cloud acquisition). Since the space is divided equally, it is suitable for the purpose of 3D thermal model reconstruction of indoor environments, as it allows for thermal inspection of the space segments. In order to create iso-surfaces realistically, it is important to calculate precise iso-values. This is typically done by ball pivoting [5], Poisson surface reconstruction [27] and

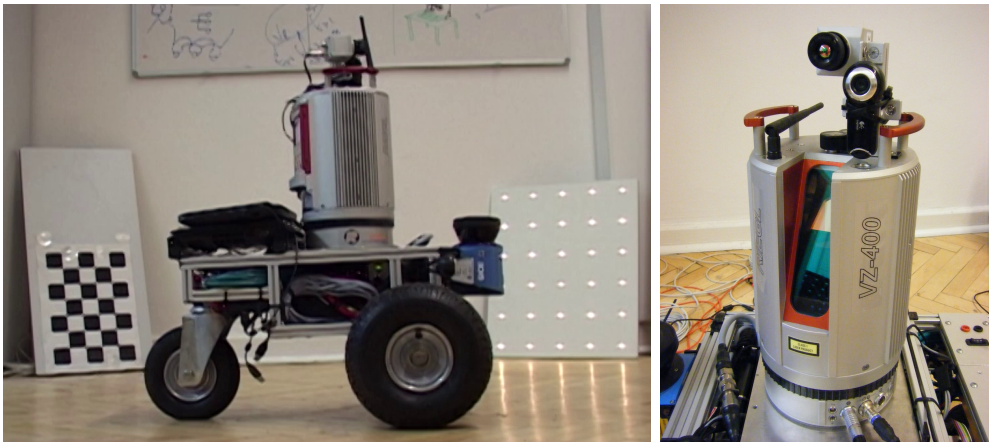


Figure 2: The robot Irma3D (left). The laser scanner with the thermal camera and the webcam mounted on top (right).

butterfly subdivision of surfaces [33]. These methods require very high resolution point clouds resulting in large data sets (millions of points) and are generally very slow.

3. Advanced Mutual Calibration between the 3D Sensor and the Thermal Camera

3.1. Experimental Setup and Data Acquisition

The setup for simultaneous acquisition of 3D laser scan data and thermal images is the robot Irma3D (see Figure 2). Irma3D is built of a Volksbot RT-3 chassis. Its main sensor is a Riegl VZ-400 laser scanner from terrestrial laser scanning. A thermal camera is mounted on top of the scanner. The optris PI160 thermal camera has an image resolution of 160×120 pixels and a thermal resolution of $0.1\,^{\circ}\mathrm{C}$ in a spectral range of 7.5 to $13\,\mu$m. It acquires images at a frame rate of 120 Hz and with an accuracy of $2\,^{\circ}\mathrm{C}$ with a field of view of approximately $40^{\circ} \times 64^{\circ}$. For acquisition of color data a Logitech QuickCam Pro 9000 webcam is used featuring a video resolution of 1600×1200 pixels. The laser scanner acquires data with a field of view of $360^{\circ} \times 100^{\circ}$. To achieve the full horizontal field of view the scanner head rotates around the vertical scanner axis when acquiring the data. We take advantage of this feature when acquiring image data. Since the cameras are mounted on top of the scanner, they are also rotated. We acquire 10 images

per camera during one scanning process to cover the full 360°. To avoid blurring and the problems that come from the necessity of synchronization we refrain from taking the images while scanning. Instead we perform a full 360°rotation for scanning and rotate back with stops at the image positions. A further advantage of this strategy is that the cameras can be connected with regular USB cables because the cable is unwound after each rotation. For obstacle avoidance a SICK LMS100 2D laser scanner is installed at the front of the robot.

3.2. Data Processing Procedure

After acquiring the 3D data it has to be merged with the image information. This processing consists of five steps that will be explained in this section.

3.2.1. Intrinsic Calibration of Thermal and Optical Camera

Each sensor perceives the world in its own local coordinate system. To join the perceived information we need the specific parameters of these coordinate systems. Each camera has unique parameters that define how a point (X, Y, Z) in world coordinates is projected onto the image plane. These parameters are calculated through a process known as geometric camera calibration. Given the focal length (f_x, f_y) of the camera and the camera center (c_x, c_y) image coordinates (x, y) are calculated as:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X/Z \\ Y/Z \\ 1 \end{bmatrix}. \tag{1}$$

Given the radial distortion coefficients k_1, k_2, k_3 and the tangential distortion coefficients p_1, p_2 and $r = \sqrt{x^2 + y^2}$ the corrected image points (x_c, y_c) are calculated as

$$\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \begin{pmatrix} x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 y + p_2(r^2 + 2x^2) \\ y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1(r^2 + 2y^2) + 2p_2 x \end{pmatrix} \tag{2}$$

To determine the parameters of optical cameras chessboard patterns are commonly used because the corners are reliably detectable in the images. A number of images showing a chessboard pattern with known number and size of squares are recorded. In each image the internal corners of the pattern are

detected. The known distances between the corner points in world coordinates allow to formulate equations (1) and (2) as a non-linear least squares problem and to solve for the calibration parameters [10].

For low resolution thermal cameras a chessboard pattern is error-prone even after heating it with an infrared lamp. For pixels that cover the edge of the squares the temperature is averaged over the black and white parts thus blurring the edges. Luhmann et al. [30] have explored the calibration procedure using different types of thermal cameras. Generally an object with a unique pattern of distinct targets is used which eases labeling and increases accuracy of the calibration process. The points are actively or passively heated. In case of passive heating different materials cause the pattern to show up. Luhmann et al. developed a pattern consisting of targets of self-adhesive foil on an aluminum plate. While the targets emit radiation based to their own temperature the reflective metal surface reflects the cold temperature of space thus leading to a strong contrast. Unfortunately this concept is not applicable for the co-calibration of the thermal camera and a laser scanner as it is very difficult to position the board in a way that the sky is reflected without occlusions and the board is completely visible in the laser scan. Instead we suggest a pattern with clearly defined heat sources such as small light bulbs as it shows up nicely in thermal images.

Figure 2 shows our pattern in the background. It is composed of 30 tiny 12 Volt lamps, each with a glass-bulb diameter of 4 mm. The overall size of the board is 500 mm (width) $\times$ 570 mm (height). Identifying the heat sources in the image enables us to perform intrinsic calibration in the same way as for optical cameras. The approach is similar to the approach used by Ham and Golparvar-Fard [23]. The main difference comes from the ability of the optris PI160 thermal camera to output the raw temperature information for each pixel rather than providing a color coded image only. While the color settings have to be carefully tuned to allow even for manual detection of the light bulbs, the raw temperature information can easily be used to detect the calibration pattern automatically in the data. A thresholding procedure is applied to create a binary image showing regions of high temperature. A further thresholding step discards effectively all regions that are too big or too small. If the remaining number of regions is equal to the number of light bulbs in the pattern the regions are sorted according to the pattern to allow for easy determination of correspondences. To calculate the exact center of the features, the mean is calculated by weighing all the pixels in the region by their temperature values.

3.2.2. Extrinsic Calibration – Cameras and Laser Scanner

After calculating the internal parameters of the cameras we need to align the camera images with the scanner coordinate system, i.e., extrinsic calibration. The three rotation and three translation parameters are known as the extrinsic camera parameters and define the geometric relation between camera and laser scanner. Once all the points are in the camera coordinate system the projection to the image can be defined up to a factor s using equation (3) [10]:

$$s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3}$$

Suppose there are n images of the calibration pattern and m planar points on the pattern considering the distortions as independent and identically distributed noise, then the maximum likelihood estimate of the transformation between the scanner and camera coordinate system is obtained by minimizing the reprojection error

$$\sum_{i=1}^{n} \sum_{j=1}^{m} ||\mathbf{p}_{ij} - \hat{\mathbf{p}}(\mathbf{A}, \mathbf{D}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{P}_j)||^2 \tag{4}$$

where $\mathbf{R}_i$ is the rotation matrix and $\mathbf{t}_i$ the translation vector of the ith image. $\mathbf{A}$ is the intrinsic matrix and $\mathbf{D}$ contains the distortion parameters as calculated in the intrinsic camera calibration. $\hat{\mathbf{p}}(\mathbf{A}, \mathbf{D}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{P}_j)$ defines the projection of point $\mathbf{P}_j$ in image i according to equation (3) and (2), and $\mathbf{p}_{ij}$ describes the pixel coordinates of the point in the image. This approach assumes that we have a number of points that are identifiable in both the laser scan and the image. For this purpose we attach the calibration pattern onto a board. For the optical camera this is a printed chessboard pattern and for the thermal camera light bulbs are arranged in a regular grid pattern. The calibration patterns are depicted in the background of Figure 2. The positions of the points in these patterns are known. Algorithm 1 detects the points in a laser scan.

To facilitate the detection of the calibration board in the point cloud data and to enable the easy positioning at different locations, the board is mounted on a tripod. This way the board hangs almost freely in the air. After removing the floor, the ceiling and most objects behind the board with a

Algorithm 1 Calibration pattern detection in a laser scan.

Require: point cloud, specification of calibration pattern

1: discard points outside the area of the expected board
2: find the most prominent plane using RANSAC (RANdom SAmple Consensus) [19]
3: project a generated plane model into the center of the detected plane
4: use ICP (Iterative Closest Point) algorithm [6] to fit the plane model to the data points
5: **if** each point from the plane model has a corresponding point in the point cloud **then**
6: **return** position of the light bulbs according to ICP result
7: **end if**

simple thresholding technique, the board becomes the most prominent plane in the data and can be detected using the RANSAC algorithm [19]. A plane model is generated by subsampling points on a plane with the dimensions of the calibration board. This model is transformed towards the center of the detected plane facing the same direction as the plane. To fit the model perfectly to the data the ICP algorithm [6] is used, thus giving the exact pose (position and orientation) of the calibration board. Since the positions of the light bulbs or chessboard corners on the board are known, their exact positions in 3D space can be calculated from the pose of the board.

3.2.3. 3D to 2D Projection and Color Mapping

During the data acquisition phase laser scans and images are acquired simultaneously. After determining the relations between scanner and cameras in the calibration step this relation is used directly to assign temperature and color values to the point cloud.

3.2.4. Projection/Occlusion/Resolution Errors

Due to the different fields of view, the sensors see different parts of the world. An area that is visible for one sensor might be occluded for the other sensor. When mapping the thermal information to the point cloud this causes wrong correspondences and therefore faulty assigned values. This impact is fortified by the low resolution of the thermal camera. With only 120 by 160 pixels per image each pixel corresponds to many 3D points seen by the laser scanner leading to errors at edges. Consequently small calibration inaccu-

racies have a large impact on the results. To solve this problem initially a fast procedure was implemented. All points that were projected onto one pixel and its neighboring pixels were clustered depending on their distance to the scanner. Assuming that most points fall onto the correct pixel a heuristic based on distance to the 3D scanner and size of the cluster determines which points are considered and enhanced with thermal information [8]. The method works sufficiently for the low resolution of the thermal camera but fails for higher resolution cameras. Therefore we now perform a ray tracing procedure that checks whether a point in the point cloud can be seen by the camera. We connect the point $\mathbf{P}$ and the camera position $\mathbf{C}$ with a straight line PC and select all points with a distance less than a threshold t to PC, i.e., all points $\mathbf{O}_i$ for which

$$|\mathbf{P} - \mathbf{O}_i|^2 - \frac{|(\mathbf{P} - \mathbf{O}_i) \cdot (\mathbf{P} - \mathbf{C})|^2}{|\mathbf{P} - \mathbf{C}|^2} < t^2 \tag{5}$$

holds true. If any point O_i lies between $\mathbf{P}$ and $\mathbf{C}$, $\mathbf{P}$ is not visible from the camera and is therefore discarded. The threshold t accounts for small inaccuracies in the calibration and the low resolution of the camera simultaneously. To speed up the checking procedure the points are organized in a kD-tree data structure. With a quick check those voxels are immediately discarded that are not traversed by the ray and therefore all the points within are ignored.

3.2.5. Scan Registration

Laser scans acquired at different positions are registered into one common coordinate system using 6D SLAM from *3DTK – The 3D Toolkit* [38]. It calculates a high presicision estimate of the scanner pose with 6 degrees of freedom. The scan registration algorithm implemented in 3DTK is described in detail in [37].

4. Sensor Placement Planning

The sensor placement planning module enables the full autonomous exploration in order to acquire a dense model of the environment. An algorithm based on the 2D horizontal plane in the point cloud at a specific height can be used efficiently for many tasks. However, this is not sufficient for creating a complete 3D model. Consider an empty room without obstacles. In a 2D

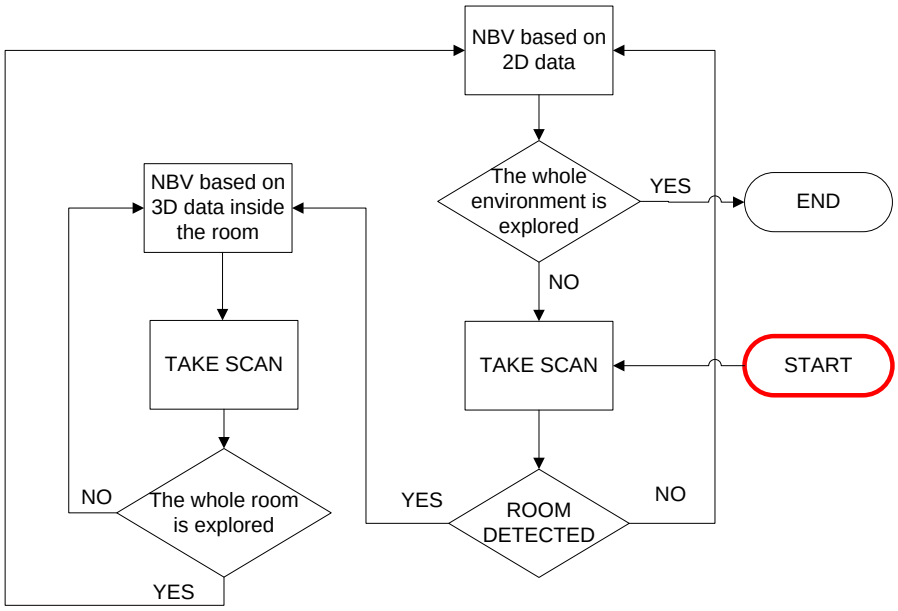


Figure 3: Block scheme of the sensor placement planning algorithm

scenario a robot with a horizontal field of view of 360° will finish after one scanning position because it perceives all the walls of the room from any position within the room. Constraints in the vertical field of view of the scanner, that cause missing data at the floor and the ceiling or parts of the walls outside of the considered height level, will simply be ignored. This is even more eminent in complex shaped rooms or when obstacles, like chairs, tables, wardrobes, etc. are present in the room and occlude other objects at varying height. Consequently, if the aim is to have a complete 3D model, a method that searches for unexplored areas in 3D is needed.

Implementing and applying a complete 3D sensor placement planning algorithm in a large indoor environment needs significant memory space for storing the occupancy information of the whole environment. One would also need to store the exploration status of each part of the environment. All together this stretches the memory demands a lot. Additionally, the computational effort needed to process all the stored information will be high and increases the exploration time. The approach proposed in this paper combines 2D and 3D planning to enable tracking of the three dimensional information of the environment with lower computation and memory demands.

The block scheme of the overall sensor placement planning algorithm is depicted in Figure 3. The main idea relies on a typical indoor environment structure, composed of enclosed spaces like rooms, halls, corridors and so on. Starting with exploration based on only 2D measurements and following the next best view (NBV) positions obtained from 2D planning, the robot detects an enclosed space, i.e. a room, and models it. At that moment the procedure of searching for the NBV position switches from 2D to 3D NBV planning, which takes into account the whole 3D environment information captured by the 3D laser scanner. The NBV planning algorithm based on 3D information explores only the detected room as a small unit of the large environment, thus needing to store only all the 3D information of this small part. Therefore, a combination of 2D and 3D information based exploration keeps the computational and memory requirements low, because only a small part of 3D model is used while exploring. When the detected room is explored, the 3D NBV planning algorithm terminates and exploration continues again with the 2D NBV planning until a new unexplored enclosed space is detected and the algorithm switches back to 3D NBV planning. The algorithm terminates when the 2D NBV algorithm detects that the whole environment is explored, i.e. when there is not any so-called jump edge left in the memory. A jump edge is an edge that separates explored and unexplored regions of the environment. The next sections describes the three main modules of the algorithm: 2D NBV planning, room detection and 3D NBV planning.

4.1. 2D NBV Planning Algorithm

The 2D NBV planning algorithm is based on our previous approach presented in [1]. The algorithm does not require any information about the environment beforehand. Initiated with a blank map it starts to explore the environment based on the first scan.

The inputs are range values uniformly distributed on the 360° field of view. The ranges are extracted from the 3D point cloud so that all range data lies in the plane parallel to the floor plane. To ensure that the robot does not hit any obstacles, a slice covering the entire height of the robot is used to create the map, i.e., all values that are between approximately 30 and 70 cm above the ground are used to create a 2D floor plan. The relatively large distance to the floor was chosen to account for small inaccuracies in the leveling of the robot.

Assuming that the environment model is initially unknown it is incrementally built after each scan. The model is hierarchical with three abstraction

levels. At the lowest level the grid map is used to store the static and dynamic obstacle information needed for path planning and obstacle avoidance. The next abstraction level contains the polygonal representation of the environment which stores environment edges, such as walls and other obstacles, which have been extracted from the range data and *jump edges* – edges that separate explored and unexplored regions of the environment. The most abstract level contains scanning position candidates which are considered for finding the NBV scanning position, i.e., the next goal position for the path planning module. We assume a setup where the robot localization problem is solved. The GMAPPING module under ROS [52] is used in our experiments. While exploring, the robot has to navigate between scanning positions in an unknown environment. We use a motion planning algorithm based on the D* algorithm and the Dynamic Window obstacle avoidance algorithm described in [47].

The 2D NBV planning algorithm is composed of three consecutive steps, which are executed at each scanning position: (1) vectorization – extracting lines from range data, (2) creation of the exploration polygon EP – building the most recent model of the environment, and (3) selection of the NBV sensor position – choosing the next goal for the path planning module. These three steps are explained in the following.

4.1.1. Vectorization

The main goal of the vectorization step is to obtain line segments from the input range data extracted from the 3D scan using the least squares method. First, the Progressive Probabilistic Hough Transform (PPHT) from the OpenCV library [26] is applied to the range data to calculate an initial estimation of the line segments, which are then used to group all range data around the calculated line segments according to their distance $\Delta\rho_k$ from the lines. Second, more precise line parameters are calculated by a least squares line fitting algorithm.

4.1.2. Creation of the Exploration Polygon

A polygonal representation of the environment is used for selecting the NBV position, for creating the gridmap, and for path planning. The calculated line segments from the vectorization step form the measurement polygon as follows. The ending points of adjacent detected lines are connected with *jump edges* and define the polygon P_i at the ith scanning position p_i. The result is a polygon P_i that is composed of real line segments and artificial

edges, i.e., jump edges, between them. However, some jump edges from the new scan might fall into the already explored area from previous scans. To discard those jump edges we use the union of the new polygon P_i (from the last scan) and the old exploration polygon EP_{i-1} (from previous scans) as a representation of the currently explored area and we discard jump edges within the union of the polygons (from GPC library [55]). In each step i, EP_i is updated as $EP_i = EP_{i-1} \bigcup P_i$. The union of two polygons keeps only those edges from both polygons that are most distant from the point of view of the robot, and thus ensures that new jump edges are not created within previously explored regions. Jump edges that are longer than the preset value Δr are considered for the selection of the next scanning position. The minimal length of the jump edge Δr is chosen in accordance with the robot dimensions and ensures that small jump edges are discarded, i.e., those the robot cannot pass through. If the non-empty extended polygon EP_i contains no reachable jump edge, it is considered as the reliable polygonal description and the exploration process stops.

4.1.3. Selection of the Next Best View Scanning Position

We use a simple heuristic criterion for selecting the NBV position similar to [18]. By taking a scan directly in front of the jump edge it is easy to imagine that we will gain a larger amount of new information than by scanning further away from the jump edge inside of the explored area. Therefore, one candidate scanning position is assigned to each jump edge. It is an obstacle free position near the mid point of the jump edge at distance d from the jump edge. d is chosen to be equal to the dimensions of the robot to ensure safety in case the jump edge is close to an obstacle that has not yet been detected. Additionally, d must be larger than the minimal sensor range. The next sensor position is chosen by maximization of a criterion that estimates the amount of unexplored regions seen from each potential position.

Figure 4 shows two candidate positions p_1 and p_2 with jump edges denoted by red lines. The current scanning position is at R. The measure of the size of the unexplored region that is possibly seen from the k-th candidate position is calculated from the angles in the triangles that are defined by the k-th candidate position and all jump edges. To maximize the information gain all jump edges have to be considered that are visible from the candidate position. In Figure 4 both jump edges are visible from p_1. Considering also the length d_j of the shortest path from the path planning module between the current robot position R and the j-th candidate position the selection

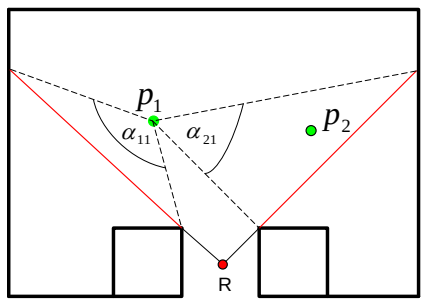


Figure 4: Selection criterion based on angles of visibility.

criterion is as follows:

$$I_j = k_1 \frac{1}{d_j} + k_2 \sum_{i=1}^{N} \alpha_{ij}. \tag{6}$$

N is the number of candidate positions and α_{ij} is the angle in the triangle defined by the j-th candidate position and the i-th jump edge. Two parameters k_1 and k_2 are used as weighting parameters of angle and distance estimations, respectively. Numerous experiments in simulation and with the real robot showed good performance with k_1 set to the maximal range distance and k_2 set to $1/N$ which averages over all angles.

4.2. Room Detection

The crucial step of the proposed sensor placement planning algorithm is the room detection algorithm, which is used to switch between the 2D and 3D NBV planning algorithms. After each scan taken at the position chosen by the 2D NBV planning algorithm, the room detection algorithm searches for the enclosed space (room) based on the current information captured from previous scans. If the room is detected, the 2D NBV algorithm is paused and the 3D NBV algorithm starts.

In the 2D exploration phase a 2D cut of the environment is used that represents the area that is traversable by the robot. For room detection such a slice close to the ground is not suitable because objects occlude the room boundaries and windows and doors interrupt them. The main idea for room detection is grounded on the detection of a closed space in the 2D polygonal line map of the environment obtained by vectorization of range data at the most suitable height level above the floor. The most suitable height level for room detection is an obstacle free 2D plane at the height where the room

boarders and walls are easily detected, i.e., a plane close to the ceiling. We choose it manually before the exploration starts. From the current 2D line map (taken at a suitable height level) of the workspace the room detection algorithm tries to detect a room. Let A be the the lines corresponding to real environment edges obtained from the last scan. The algorithm searches for a room starting from the first line within the set A and tries to close the loop through other lines in the entire map. The process proceeds by finding the line closest to the current line in each step. When the loop is closed, i.e., when the nearest line is the starting line again, the room is detected. The nearest line search refers to the nearest ending point of a line in the vicinity of the current line. The vicinity area around the current line is defined with the radius parameter equal to the expected doorway width. With that, we ignore holes caused by doorways and windows inside the room. If there is no room detected starting from the first line in set A, the detection process starts from the second line in A. The order of the lines in A is not relevant. In case no closed polygon can be found with any line from set A as starting line, the room could not be detected in that step and exploration continues with the 2D NBV algorithm.

4.3. 3D NBV Planning Algorithm

The room detection algorithm provides the room parameters including the boundary area and the coordinates of the room. When the room is detected at least one scan inside the room is available since the robot has already entered the detected room and taken at least one scan inside which has been used for the room detection. From the available scans inside the room the initial 3D model of the room is built and the exploration continues by using the 3D model based NBV planning algorithm. The main idea of the 3D NBV planning algorithm is described hereafter.

To obtain NBV positions the 3D model of the detected room needs to hold information of the explored and unexplored area inside the room. We use a voxel based 3D model where each voxel has one of the following labels: *occupied* if the volume within the voxel is occupied, *unseen* if the occupied status of the voxel is unknown or *empty* if the voxel is empty with no obstacles inside. The dimensions of the room define the number of voxels that should be used to cover the whole area of the room, i.e., memory allocation for the detected room. The shape of the enclosed space can be arbitrary. A cuboid 3D voxel model that encloses the entire detected room is chosen. The voxel model has to be large enough to enclose the largest room expected in the

environment. Although there could be voxels in the cuboid model which are not part of the room, they are not considered during NBV position planning.

Once the room is detected, its voxel based 3D map is initialized with all voxels set to *unseen*, since we do not have any information on the environment. The stored scans that were taken inside the detected room are used to update its initial voxel based 3D model. This is performed by using the ray tracing algorithm which traces a ray from the position where the scan was taken to each data point in the scan. Each voxel that is crossed by the ray is marked as *empty* and the voxel containing the data point is marked as *occupied*. The potential NBV position candidates are all voxels at the height of the laser scanner with the status *empty*. Position candidates that are not reachable for the robot are then removed from the list. The aim is to choose the candidate scanning position from where the most *unseen* voxels could be seen. For each candidate scanning position we count the number of *unseen* voxels. A ray is traced from the candidate scanning position to each *unseen* voxel and, if all crossed voxels are *empty*, the counter is incremented. Since considering every *unseen* voxel is unnecessarily time consuming only *unseen* voxels with at least one *empty* neighbor are taken into account [7]. In that way the number of voxels that need to be tested is decreased and voxels outside the room boundaries are not considered. The approach is similar to the jump edges in 2D in the way that *unseen* voxels that are taken into account actually corresponds to jump planes which divide explored and unexplored regions. Constraints in the field of view and range properties that are limits introduced by the sensors are considered by checking the range and the angle between the candidate scanning position and the *unseen* voxels. If the constraints are not satisfied, the *unseen* voxel is not counted. After finding a location that maximizes the number of unseen voxels, i.e., the NBV position, the robot drives to it, takes the 3D scan, and the whole procedure is repeated. The algorithm stops when the number of *unseen* voxels that can be seen from the best candidate position is below some predefined threshold V_{min} and the room is considered explored.

5. Planning Results

Experiments were carried out in a research building (Figure 5) at Jacobs University Bremen, Germany with the robot Irma3D (Figure 2). The aim was to build a complete 3D model of the environment based on 3D scans with thermal information attached. Each scan took 3 minutes and 15 seconds.

(a) (b) (c)

(d) (e) (f)

Figure 5: Pictures of the environment taken by Irma3D during the experiments. The map with the scanning positions is given in Figure 6. a) and b) room 1 seen from scanning position 2; c) the corridor seen from position 6 facing towards position 4; d) looking into room 3 from position 6; the small corridor connecting room 2 and 3 seen from position 11; e) room 2 seen from position 10.

The laser scanner is constrained by a vertical field of view of 100° and the thermal camera with a vertical opening angle of 60°, respectively. Since the laser scanner field of view exceeds that of the thermal camera, the constraints of the algorithm are set to consider the field of view of the camera. The voxel

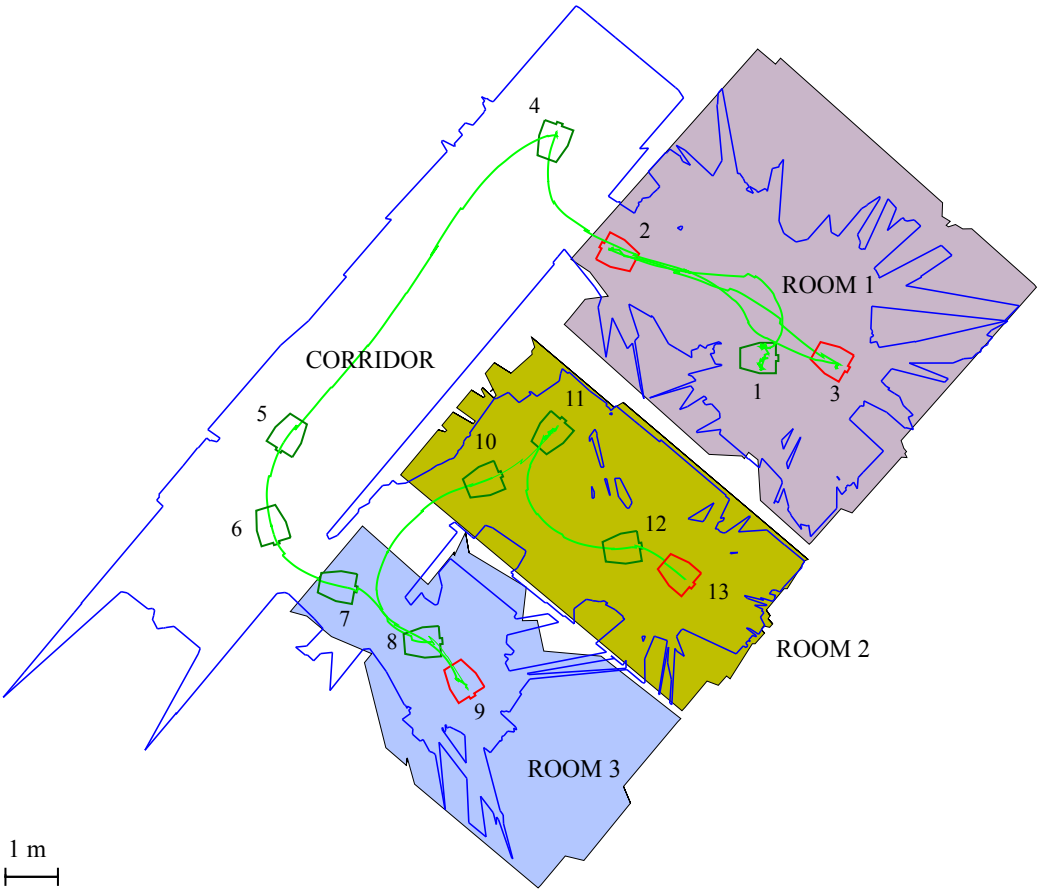
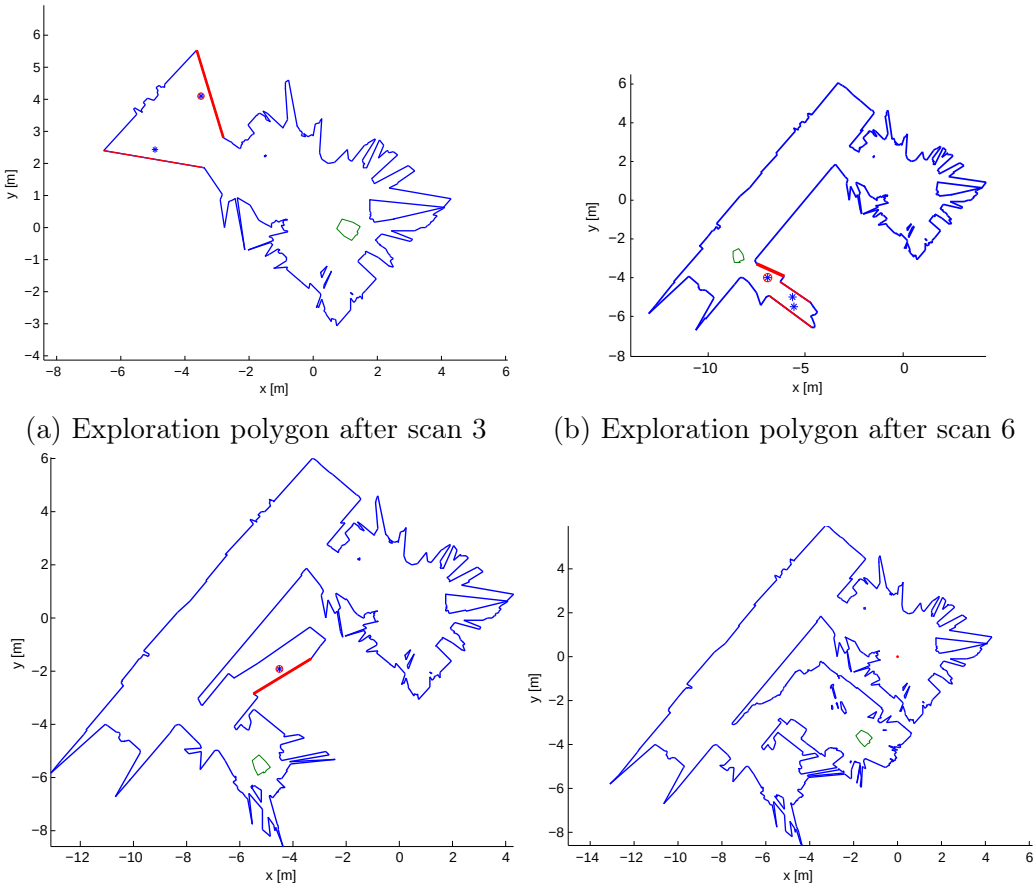


Figure 6: Robot path and scanning positions during exploration

volume used in the experiment was set to $0.008\,\mathrm{m}^3$ ($0.2\,\mathrm{m} \times 0.2\,\mathrm{m} \times 0.2\,\mathrm{m}$). For creating the 2D map a slice of the 3D scan was taken at a height between approximately 30 cm and 70 cm above the ground to make sure the robot hits no obstacles at any height. The room detection height level was set to 2.5 m to avoid difficulties with windows and doors. Figure 6 to 8 present the results of an experiment illustrating the behavior of the proposed sensor placement planning algorithm.

Figure 6 shows the floor plan of the explored environment which consists of three rooms and a corridor. The scanning positions chosen by the proposed sensor placement scanning algorithm are ordered from the start position (position 1) to the end position (position 13) and marked with the



(a) Exploration polygon after scan 3

(b) Exploration polygon after scan 6

(c) Exploration polygon after scan 9

(d) Exploration polygon after scan 13

Figure 7: Important exploration polygons during the experiment (jump edges are in red color and 2D NBV scanning positions marked with cross characters in red circles).

robot footprints. Green footprints refer to scanning positions generated by the 2D algorithms and red footprints by the 3D NBV algorithm, respectively. Some photos illustrating the environment are given in Figure 5. The photos are excerpts from the data that the robot Irma3D acquired during the experiment. The robot started from position 1 in room 1 and took the initial scan. The first scan was sufficient for the room detection algorithm to detect room 1. When the room was detected, the jump edges inside room 1 were discarded, leaving only the jump edges detected outside of room 1

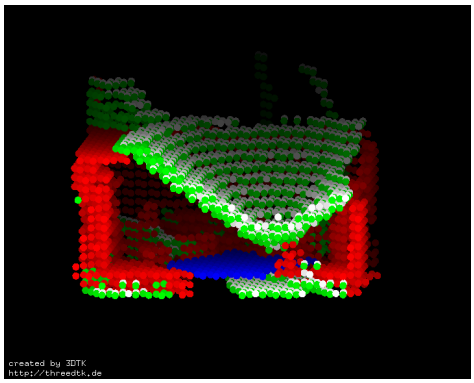
for continuing with the 2D NBV planning algorithm after exploring room 1. The initial 3D voxel model of the room was built and the 3D NBV planning algorithm was switched on. Then the 3D NBV planning algorithm chose position 2, took the scan there and updated the 3D model of room 1. Since the number of unseen voxels was larger than the threshold ($V_{min} = 15$ voxels) the 3D NBV algorithm continued with the exploration of room 1 and chose scanning position 3. After the third scan, the 3D model of room 1 was of the desired accuracy and the exploration was continued with the 2D NBV algorithm, which moved the robot to position 4 in the corridor, i.e. in front of one of the two available jump edges (see Figure 7a). The 2D NBV algorithm chose two additional scanning positions in the corridor (positions 5 and 6). The reason for the jump edge at position 5 is a range constraint of 8 m on the laser data in 2D causing part of the corridor not to be seen in the 2D laser data from position 4. Further away from the robot the points are very sparse making line extraction difficult. After the corridor, the robot moved to position 7 at the entrance of room 3, i.e. to the edge with the lowest value of the criterion (Figure 7b) without exploring the corridor with the 3D NBV algorithm. The corridor was not considered as a room since it was closed on both sides with glass doors as can be seen in Figure 5(c). At the room detection height level of 2.5 m the glass doors were not visible in the laser scan.

Due to the lack of clutter in the corridor three scanning positions were sufficient to satisfy the criteria of the 2D NBV algorithm. After scan 8 room 2 was detected and fully explored with only one additional position from the 3D NBV algorithm. Afterwards, 2D NBV planning chose position 10 based on the polygon shown in Figure 7c and position 11 and 12 before room 3 was detected. Finally, position 13 was chosen by the 3D NBV algorithm to achieve the required accuracy of the room model. The exploration of the environment finished here as there were no jump edges in the memory (Figure 7d), which means that the whole environment was explored and modeled.

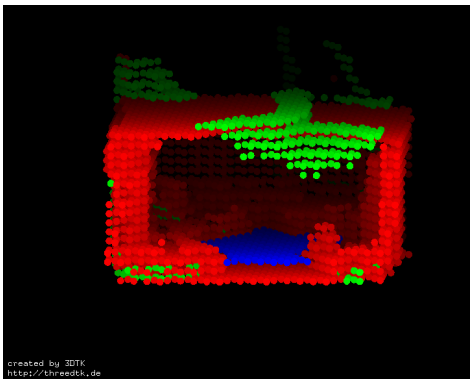
Figure 8 represents the 3D voxel models of room 1 after the first (8a) and the third (8b) scan. The *occupied* voxels are colored red, potential position voxels (PP) are colored blue, while *unseen* voxels are colored green. In the figure we also have white voxels, which are unseen voxels that can be seen from at least one PP position in the workspace. They are a subset of the *unseen* voxels and are treated as such in the exploration algorithm. They only serve to show that some of the *unseen* voxels could never be seen from any position. The specific cone in the figure is a consequence of the sensor

Table 1: The number of *unseen (U)*, *occupied (O)* and the number of *unseen* voxels that can be seen from the best next position in 3D (*UB*) at each scan inside room 1

Voxel type	Scan 1	Scan 2	Scan 3
O (red)	4091	6207	7034
U (green)	2104	1248	1207
UB	965	203	0



(a) Initial model of room 1 after the first scan



(b) Final model of room 1 after the third scan

Figure 8: 3D voxel based models of room 1. *occupied* voxels are colored red, potential position voxels (*PP*) are colored blue, *unseen* voxels are colored green. White voxels are unseen voxels that can be seen from at least one *PP* position in the workspace.

field of view constraints. The other *unseen* (green) voxels are situated under the tables, chairs and other obstacles inside the room representing unexplored area. As can be seen in the model and in the Table 1, the number of *unseen* voxels that can be seen from the best position is zero after the last scan inside room 1.

To evaluate the benefits of the 3D exploration strategy, we conducted two new experiments in room 1, one using only 2D exploration and one with both 2D and 3D exploration. We placed the robot at the same starting position in both experiments. The results are seen in Figure 9 and 10. Figure 9 shows

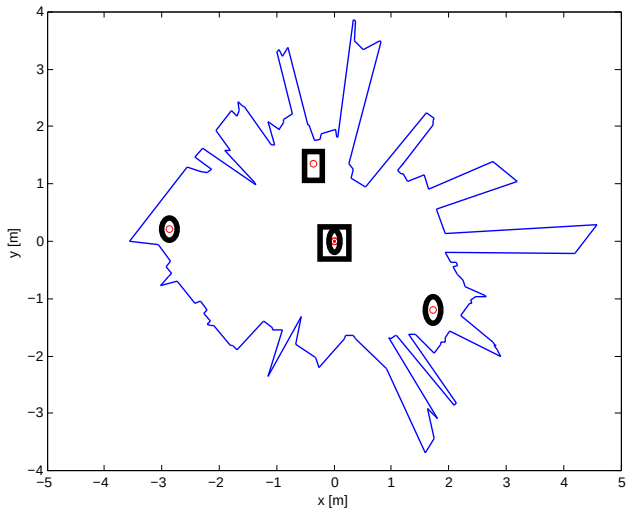


Figure 9: Comparison of 2D and 3D NBV algorithm

the final exploration polygon with the positions where the scans were taken. Rectangular marks show positions chosen only by 2D NBV and circles refer to 3D NBV based positions. In the 2D mode the robot finished after two scanning positions. As can be seen in the panorama image, large parts of the ceiling, the floor and the walls are not captured with the thermal camera. The unexplored area is drastically reduced when 3D exploration is employed as well. From Table 2 it becomes clear that 703 *unseen (UB)* voxels could still be seen from the NBV position after the 2D exploration.

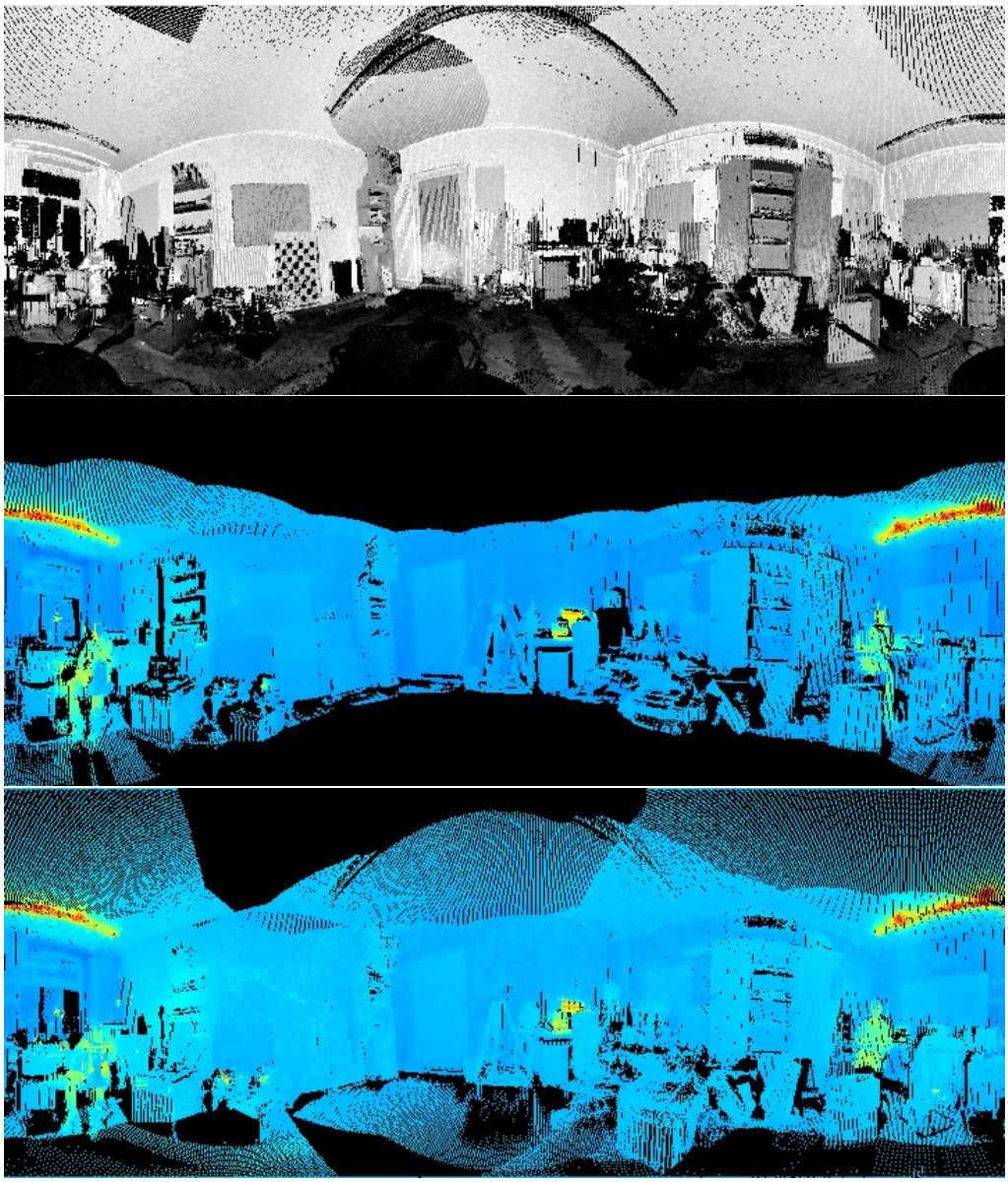


Figure 10: Panorama images of room 1. Top: The entire room with reflectance values as captured by the laser scanner using 3D exploration. Middle: The part captured with thermal information using 2D exploration. Bottom: The part captured with thermal information using 3D exploration. Note that the thermal camera has a drastically reduced field of view as compared to the laser scanner.

Table 2: The number of *unseen (U)*, *occupied (O)* and the number of *unseen* voxels that can be seen from the best next position in 3D (*UB*) at each scan inside room 1 after 2D and 3D exploration.

	3D NBV			2D NBV	
Voxel type	Scan 1	Scan 2	Scan 3	Scan 1	Scan 2
O (red)	4068	5321	6106	4033	4958
U (green)	2275	1524	1192	2320	2036
UB	1025	361	0	1017	703

6. 3D Thermal Surface Model Building

The complete model of the environment can be inspected in the viewer from 3DTK enhanced with either reflectance values or thermal data (see Figure 1). The color scale is adjustable for a good view of the temperature distribution in the current data. In all images depicted here, blue corresponds to cold temperatures while dark red corresponds to warm temperature values. Switching between the different views enables the user to detect sources of wasted energy and to locate them clearly in the 3D view. The Riegl VZ-400 laser scanner has an accuracy of 5 mm [32]. In previous work it was shown that the scan matching algorithm used here leads to a positional error of less than 4 cm when joining several 3D laser scans even in large outdoor environments [37]. However, humans are used to virtual models consisting of filled structures without holes, i.e., surfaces or meshes. The process of generating surfaces from a set of points is called surface reconstruction. We present a method to reconstruct a 3D model with added thermal field information and to retrieve information from the finished model. The automatic detection of heat sources in the model to point out possible leaks is of special interest.

6.1. Applying the Marching Cubes Algorithm

The Marching Cubes Algorithm (MCA) calculates meshes based on iso-surfaces from point clouds that are organized in voxels. It calculates triangles that separate each voxel into the parts that lie within and outside of the object that is to be reconstructed. To generate the iso-surfaces more quickly, a probabilistic approximation method is used [46]. Every point in the input

point cloud is assigned a Gaussian distribution. The final iso-surface is calculated by applying an aggregation operator, i.e., minimum, maximum, or sum. Changing the width of the Gaussian bell reduces the influence on the neighboring points/voxels. This method improves the results if normals in every point are provided.

6.2. Temperature Scalar Field Mapping

The Gaussian estimation of iso-values and the subsequent MCA reconstruction do not deal with scalar temperature values acquired by the thermal camera. These values have to be mapped onto the reconstructed surface using an appropriate color scale.

The surface is reconstructed from the point cloud. The temperature values are assigned to the points. Mapping the temperature color distribution onto the reconstructed surface is difficult because not every point on the surface corresponds directly to one point in the original point cloud and therefore cannot be assigned a temperature/color value directly. To overcome this problem, the points from the model, i.e., the vertices of the triangle mesh are organized in a spatial data structure that enables fast range and nearest neighbor (NN) searches. For each point in the point cloud the closest point in the mesh is found and assigned the scalar temperature value. R-trees have proven to be an efficient data structure for solving the scalar field mapping problem [39], but the original kD-tree based solution produces the model with better quality and scalar field mapping precision. An R-tree organizes spatial objects by defining bounding boxes that contain all objects indexed by the child nodes. In the case of scalar field mapping the bounding box of leaf nodes contains preferably one or more points that are vertices of a polygonal face. The size of these bounding boxes influences the precision of the NN search result and thus the overall quality of the scalar field mapping as shown in [39].

6.3. Improving the general precision of the model reconstruction

One of the major drawbacks of the presented reconstruction and scalar field mapping is the precision. The precision of the reconstructed model is mainly affected by the density of the point cloud [41]. In our case, the average density of the point cloud is greatly affected by outliers that increase the size of the bounding box of the model. These outliers are often noise caused by reflections from windows or simply objects that are seen through the window and are not of interest for the model.

This corresponds to a typical point cloud segmentation problem, where the point cloud is to be segmented into several smaller parts. The task given here differs from other segmentation problems as we can assume that valid points and invalid points will be somewhat disconnected. Thus, a connectivity criterium based on K-means clustering effectively eliminates points that are problematic for the reconstruction of the 3D model without the need for time and resource consuming segmentation algorithms. A choice of $K = 4$ has shown to give good results in the experiments. In general, it is not easy to conclude the perfect number of clusters from the data. Standard techniques for choosing the initial number of clusters are much slower and tend to choose a number much higher than necessary, thus removing many of the valid points [41].

6.4. Detection of thermal features in the model

Heat sources are identified by analyzing the scalar field values contained in the 3D model of an indoor environment based on a high local increase of temperature. In buildings, sources of heat are radiators with pipes, air conditioners, incandescent lights and other electrical equipment emitting high amounts of infra-red radiation. We observe from Figure 11 that computer monitors and light bulbs are major sources of heat. The detection can be considered as a typical image processing task, but in our case a full 3D model (or at least 3D point cloud) makes the task more complicated. In [42] we analyze scalar value thresholding, K-means clustering and Fuzzy c-means. The scalar value thresholding method from [49] was modified so that all vertices in the mesh, with a temperature value succeeding a threshold, are marked as potential heat sources. K-means and Fuzzy c-means cluster the points from the model based on their temperature values. In [42] we show that K-means shows the most promising results, with fuzzy c-means having somewhat worse performance, but with almost the same quality.

6.5. Reconstructed model

Figure 12 shows the reconstruction of one part of the scanned indoor environment, a scan taken in room 1. The model contains close to 11,926,000 points for a volume of less than $200\,\mathrm{m}^3$. Compared to the data sets presented in [5], [27] and [33], this data set is rather small and the expected precision is lower. However, it is clear that geometry is preserved, although the precision when it comes to reconstructing furniture, computer monitors, and people is not sufficient. To increase precision, a larger space subdivision

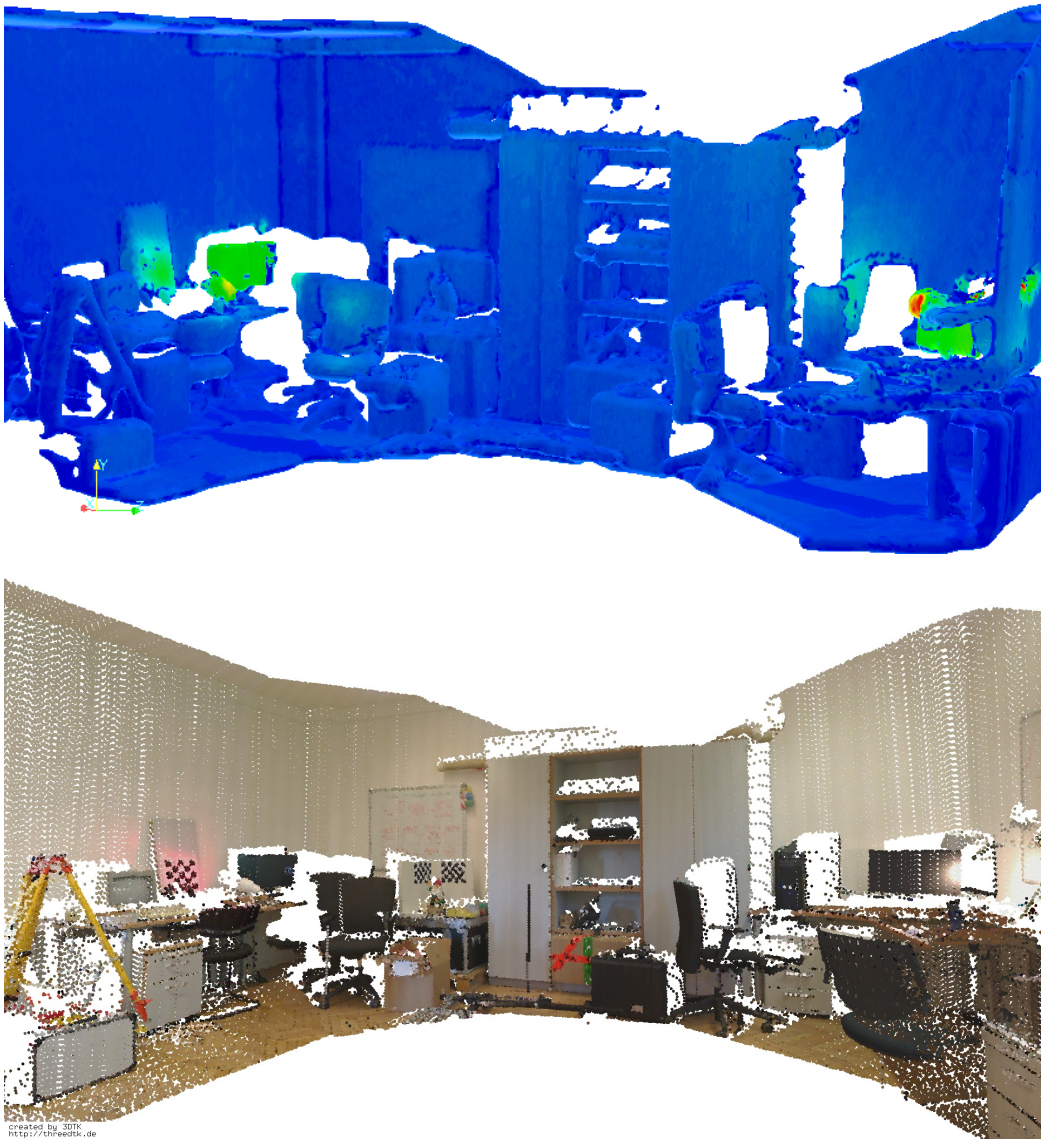


Figure 11: Top: Reconstructed 3D model of an office with furniture, lamps, computers etc. Bottom: Colored point cloud of the same office.

is needed. This comes at the expense of higher computing power and memory requirements to store the model [40]. Time performance dependence on spatial subdivision for the part of the dataset is given in Table 3.

To evaluate the obtained model visual perception of an expert is neces-

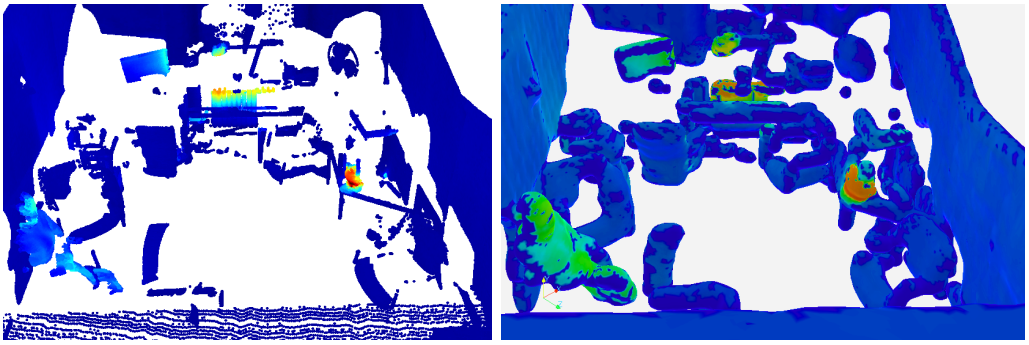


Figure 12: Image of a point cloud (left) captured at scan position 12 in office room 3 and the reconstruction (right)

Figure 13: Reconstructed 3D model of an office (room 1 from Figure 6); Left: the original point cloud divided into four clusters, one with valid points (green) and three with problematic points that were seen through the window and are thus far away from the building that is supposed to be modeled; Right: the resulting point cloud after the removal of the three invalid clusters. It is quite clear that the volume of the cuboid containing the entire model is drastically decreased, thus allowing for higher precision in the model reconstruction. At closer look, one sees that besides the office room (marked with green) and some parts of the hallway (yellow) there are still some outliers left that were seen through the window (red). However, their influence on the volume of the bounding box is minor compared to the removed clusters.

sary [40]. The methodology presented in section 6.3 attempts to minimize the local error. We managed to increase the point cloud density by the factor of ≈ 50. In the end this decreases the reconstruction error to a few centimeters compared to the original 10-20 cm [41]. In Figure 13 the benefit of removing

Table 3: Spatial subdivision influence on time performance of the reconstruction algorithm

Subdivision	50	100	150
Execution Time	6.2 seconds	24.9 seconds	55.1 seconds
Number of Polys	41322	195824	501229

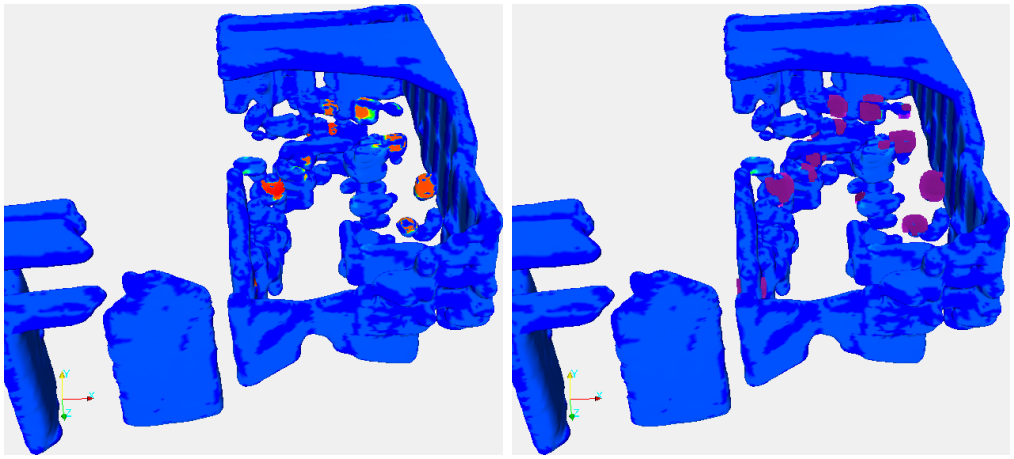


Figure 14: The left image shows the model that serves as an input for the detection algorithm; the right image shows potential heat sources labeled with purple color

problematic points is clearly shown. The reconstruction algorithm takes a point cloud with much higher density as an input and produces a model with higher precision.

Finally, we present the results of the detection of heat sources in the discussed environment. There are several sources of heat in this model and they have a color ranging from orange to red. These points have the surface temperature from roughly 35 °C to 60 °C. In Figure 14 we present the input model and the output of the detection algorithm which labels the potential heat sources.

7. Conclusions

This article presents a complete system for autonomous 3D exploration and thermal mapping of an indoor environment. The data is autonomously

collected, thermal and color information is automatically mapped onto the 3D data and the environment is reconstructed into a thermal model. To perceive the entire environment we propose a model based 3D sensor placement approach that uses a voxel representation and room extraction to drastically decrease the computational requirements while still reaching high accuracy. A viewer is presented to analyze the reconstructed model afterwards and to inspect the automatically detected points of interest in the data.

The work presented here is fundamental research and does not claim to offer a ready to use product. The system is a first step towards a tool that targets at improving the effectiveness of indoor thermography. The main goal, to design a system that autonomously collects the data necessary to create a model and perform the thermal analysis of a building with this model, is achieved. The system is designed to point out possible problems in the data to an experts who interprets them with his experience and expertise. Hotspots as they appear in heat sources and power lines were chosen as an example here. The general idea works the same for cold spots. The system is more adequate for indoor office buildings or for very cluttered environments, in which the room extraction algorithm will find a closed loop in the part of the environment that is not a real room.

However, setting a predefined maximal cuboid size supports the exploration of larger rooms. If the currently collected data reaches the predefined maximal cuboid size, the algorithm will switch to the 3D phase using a cuboid part of the environment that does not exceed the allocated memory. In our future work, we plan to extend the algorithm for such cases by considering possibilities of finding virtual rooms inside a large room.

The majority of work that still has to be done lies in the post processing, the model building and the detection of thermal flaws. As the interpretation of thermal data underlies the strong influence of the material properties and the external conditions such as room temperature and humidity, it is very hard to make decisions automatically. A system, such as this, aims at pointing out points of interest to the expert. In this paper, a method is presented that aims at pointing out hot spots, as they appear in heat sources and power lines. The evaluation of the effectiveness of this approach for real thermal issues is still subject to further research. For the detection of cold spots, such as thermal bridges, it is generally possible to use the same method with a lower boundary. However, it is to be expected that the temperature differences are much lower and therefore harder to detect. Machine learning approaches could help in detecting and classifying these points of interest. Some prelim-

inary work on automated Building Information Modeling (BIM) from laser scans by detecting doors and windows have been presented [57, 53, 17]. We have extended this work by labeling windows in 3D thermal data [16]. Future work will build upon this work and further investigate the use of machine learning for interpreting thermal 3D models.

We plan to further improve the precision of the 3D model by analyzing other factors that influence low point cloud density. A more effective method for removing invalid points using associations based on connectivity might lead to better results. In this work we focused on creating the 3D thermal model. The point cloud viewer offers the possibility to switch between thermal and color mode. A good method has to be found to make the reconstructed model available with both modalities as well. Possibilities are to switch between the two models, to create a model with one modality and add the option to show the corresponding image of the other modality at a certain spot, or to overlay the color model with the thermal information, thus modifying the color space of regions with temperature peaks and keeping the true color for the other areas.

References

[1] Marija Đakulović, Sandor Ileš, and Ivan Petrović. Exploration and Mapping of Unknown Polygonal Environments Based on Uncertain Range Data. *AUTOMATIKA: Journal for Control, Measurement, Electronics, Computing and Communications*, 52(2):118–131, 2011.

[2] Enkhbayar Altantsetseg, Yuta Muraki, Fumito Chiba, and Kouichi Konno. Surface reconstruction of stone tools by using four-directional measurement machine. *The International Journal of Virtual Reality*, 2011.

[3] Nina Amenta, Marshall Bern, and Manolis Kamvysselis. A new voronoi–based surface reconstruction algorithm. In *Proccedings of the 25th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '98, pages 415–421, New York, NY, USA, 1998. ACM.

[4] Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag TELOS, Santa Clara, CA, USA, 3rd edition, 2008.

[5] Fausto Bernardini, Joshua Mittleman, Holly Rushmeier, Claudio Silva, and Gabriel Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5:349–359, 1999.

[6] Paul J. Besl and Neil D. McKay. A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239 – 256, February 1992.

[7] Paul S. Blaer and Peter K. Allen. Data acquisition and view planning for 3-D modeling tasks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 417–422. IEEE, 2007.

[8] Dorit Borrmann, Hassan Afzal, Jan Elseberg, and Andreas Nüchter. Mutual Calibration for 3D Thermal Mapping. In *Proceedings of the 10th Symposium on Robot Control (SYROCO)*, Dubrovnik, Croatia, September 2012.

[9] Dorit Borrmann, Andreas Nüchter, Marija Đakulović, Ivan Maurović, Ivan Petrović, Dinko Osmanković, and Jasmin Velagić. The Project ThermalMapper – Thermal 3D Mapping of Indoor Environments for Saving Energy. In *Proceedings of the 10th Symposium on Robot Control (SYROCO)*, Dubrovnik, Croatia, September 2012.

[10] Gary Bradski and Adrian Kaehler. *Learning OpenCV, Computer Vision with OpenCV library*. O'Reilly Media, first edition, 2008.

[11] M. Cabrelles, S. Galcera, S. Navarro, J. L. Lerma, T. Akasheh, and N. Haddad. Integration of 3D laser scanning, photogrammetry and thermography to record architectural monuments. In *Proceedings of the 22nd CIPA Symposium*, Kyoto, Japan, October 2009.

[12] Evgeni V. Chernyaev. Marching cubes 33: Construction of topologically correct isosurfaces. Technical Report CN/95-17, CERN Report, 1995.

[13] Y. Chrysanthou, K. Cain, N. Silberman, F. Niccolucci (editors, Gerhard Heinrich Bendels, Patrick Degener, Marcel KÃűrtgen, and Reinhard Klein. Image-based registration of 3D-range data using feature surface elements, 2011.

[14] Commission of the European Communities. Addressing the challenge of energy efficiency through Information and Communication Technologies. Communications from the Commisison to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, COM(2008) 241 final, May 2008.

[15] D. González-Aguilera, P. Rodriguez-Gonzalvez, J. Armesto, S. Lagüela. Novel approach to 3D thermography and energy efficiency evaluation. *Energy and Buildings*, 54:436–443, 2012.

[16] Girum G. Demisse, Dorit Borrmann, and Andreas Nüchter. Interpreting Thermal 3D Models of Indoor Environments for Energy Efficiency. In *Proceedings of the 16th Conference on Advanced Robotics*, Montevideo, Uruguay, November 2013.

[17] R. C. Dumitru, D. Borrmann, and A. Nüchter. Interior Reconstruction using the 3D Hough Transform. In *Proceedings of the 5th ISPRS International Workshop 3D-ARCH 2013: "3D Virtual Reconstruction and Visualization of Complex Architectures"*, Trentro, Italy, February 2013.

[18] A. Ekman, A. Torne, and D. Stromberg. Exploration of polygonal environments using range data. *Systems, Man, and Cybernetics, Part B: Cybernetics*, 27(2):250–255, 1997.

[19] M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24:381 – 395, 1981.

[20] FLIR Systems AB. *Thermal imaging guidebook for building and renewable energy applications.* In cooperation with the Infrared Training Center (ITC), Sweden, 2011.

[21] Nabil A. Fouad and Torsten Richter. *Leitfaden Thermografie im Bauwesen – Theorie, Anwendungsgebiete, praktische Umsetzung.* Fraunhofer IRB Verlag, Stuttgart, 4th edition, 2012.

[22] H. González-Baños and J.C. Latombe. A randomized art gallery algorithm for sensor placement. *SCG'01: Proceedings of the Seventeenth Annual Symposium on Computational Geometry*, pages 232–240, 2001.

[23] Youngjib Ham and Mani Golparvar-Fard. An automated vision-based method for rapid 3D energy performance modeling of existing buildings using thermal and digital imagery. *Advanced Engineering Informatics*, 2013.

[24] L. Högner and U. Stilla. Texture extraction for building models from ir sequences of urban areas. In *Proceedings of the 2007 Urban Remote Sensing Joint event: URBAN / URS*, 2007.

[25] D. Iwaszczuk, L. Hoegner, and U. Stilla. Matching of 3D building models with IR images for texture extraction. In *Joint Urban Remote Sensing Event*, Munich, Germany, April 2011.

[26] J. Matas and C. Galambos and J. Kittler. Progressive Probabilistic Hough Transform. In *Proceedings of the British Machine Vision Conference*, volume 1, pages 256–265, 1998.

[27] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, SGP '06, pages 61–70, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.

[28] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21:163–169, August 1987.

[29] K.L. Low and A. Lastra. Efficient constraint evaluation algorithms for hierarchical next-best-view planning. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pages 830–837. IEEE, 2007.

[30] Thomas Luhmann, Johannes Piechel, Julia Ohm, and Thorsten Roelfs. Geometric calibration of thermographic cameras. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information*, volume 38 part 5, Newcastle upon Tyne, UK, 2010. Commission V Symposium.

[31] S. Markov and A. Birk. Autonomous victim detection in rescue operations by modelbased reproduction of 3D scenes from 2D thermal images. In *KI 2007: Advances in Artificial Intelligence, Proccedings of*

30th Annual German Conference on AI(KI '07), number 4667 in LNAI, Osnabrück, Germany, September 2007.

[32] K. Meschelke. Prüfverfahren für terrestrische Laserscanner an der HCU Hamburg. In *Terrestrisches Laserscanning (TLS 2011) Beiträge zum DVW-Seminar in Fulda*, pages 69–92, Fulda, Germany, November 2011.

[33] Iklima Mohamad and Abdullah Bade. Butterfly subdivision scheme used for the unorganized points reconstruction in virtual environment. In *Proceedings of the 2009 International Conference on Computer Technology and Development - Volume 02*, ICCTD '09, pages 345–349, Washington, DC, USA, 2009. IEEE Computer Society.

[34] S. J. Moorehead, R. Simmons, and W. L. Whittaker. Autonomous exploration using multiple sources of information. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '01)*, Seoul, Korea, May 2001.

[35] Gregory M. Nielson and Bernd Hamann. The asymptotic decider: resolving the ambiguity in marching cubes. In *Proceedings of the 2nd conference on Visualization '91*, VIS '91, pages 83–91, Los Alamitos, CA, USA, 1991. IEEE Computer Society Press.

[36] A. Nüchter. *3D Robotic Mapping: The Simultaneous Localization and Mapping Problem with Six Degrees of Freedom*. Number 52 in Springer Tracts in Advanced Robotics. Springer, February 2009.

[37] A. Nüchter, J. Elseberg, P. Schneider, and D. Paulus. Linear Solutions to the Scan Registration Problem. *Computer Vision and Image Understanding (CVIU)*, 2009.

[38] Andreas Nüchter et al. 3DTK – The 3D Toolkit. `http://slam6d.sourceforge.net/`, August 2011.

[39] Dinko Osmankovic, Haris Supic, and Jasmin Velagic. Performance and quality assessment of R-tree based nearest neighbour search in the scalar field mapping technique. In *IECON 2013, 39th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2013.

[40] Dinko Osmankovic and Jamin Velagic. Reconstructing the 3D thermal model of indoor environment from unorganized data set acquired by 3D

laser scans and thermal imaging camera. In *Intelligent Control (ISIC), 2012 IEEE International Symposium on*, pages 13–18. IEEE, 2012.

[41] Dinko Osmankovic and Jasmin Velagic. Increasing the precision of reconstructed 3D model of indoor robot environment by elimination of problematic points. *Robot Control*, 10(1):594–598, 2012.

[42] Dinko Osmankovic and Jasmin Velagic. Detecting heat sources from 3D thermal model of indoor environment. In *ICAT 2013, XXIV International Conference on Information, Communication and Automation Technologies*, 2013.

[43] A. Pelagottia, A. Del Mastio, F. Uccheddu, and F. Remondino. Automated multispectral texture mapping of 3D models. In *17th European Signal Processing Conference (EUSIPCO 2009)*, Glasgow, Scotland, August 2009.

[44] S. Prakash, L. Y. Pei, and T. Caelli. 3D mapping of surface temperature using thermal stereo. In *Proceedings of International Conference on Control, Automation, Robotics and Vision (ICARCV '06)*, December 2006.

[45] Project Webpage. Project ThermalMapper. `http://www.faculty.jacobs-university.de/anuechter/thermalmapper.html`, August 2011.

[46] Will Schroeder, Ken Martin, and Bill Lorensen. *The visualization toolkit: an object-oriented approach to 3D graphics*. Kitware, 01 2006.

[47] Marija Seder and Ivan Petrović. Dynamic window based approach to mobile robot motion control in the presence of moving obstacles. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA '07)*, pages 1986–1991, 2007.

[48] Stephen Vidas, Peyman Moghadam. HeatWave: A handheld 3D thermography system for energy auditing. *Energy and Buildings*, 66:445–460, November 2013.

[49] George Stockman and Linda G. Shapiro. *Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2001.

[50] Peter Su and Robert L. Scot Drysdale. A comparison of sequential delaunay triangulation algorithms. *Computational Geometry*, 7(5–6):361 – 385, 1997.

[51] H. Surmann, A. Nüchter, and J. Hertzberg. An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Journal Robotics and Autonomous Systems (JRAS)*, 45(3–4):181–198, December 2003.

[52] ROS Robot Operating System. `http://www.ros.org`.

[53] Pingbo Tang, Daniel Huber, Burcu Akinci, Robert Lipman, and Alan Lytle. Automatic reconstruction of as-built building information models from laser-scanned point clouds: A review of related techniques. *Automation in Construction*, 19(7):829 – 843, 2010.

[54] B. Tovar, R. Murrieta-Cid, and S.M. LaValle. Distance-optimal navigation in an unknown environment without sensing distances. *IEEE Transactions on Robotics*, 23(3):506–518, 2007.

[55] B.R. Vatti. A generic solution to polygon clipping. *Communications of the ACM*, 35(7):56–63, 1992.

[56] Stephen Vidas, Peyman Moghadam, and Michael Bosse. 3D Thermal Mapping of Building Interiors Using an RGB-D and Thermal Camera. In *Proceedings of the IEEE Conference on Robotics and Automation (ICRA '13)*, Karlsruhe, Germany, May 2013.

[57] Xuehan Xiong, Antonio Adan Oliver, Burcu Akinci, and Daniel Huber. Automatic creation of semantically rich 3d building models from laser scanner data. *Automation in Construction*, 31:325–337, May 2013.