The Project ThermalMapper – Thermal 3D Mapping of Indoor Environments for Saving Energy [⋆]

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

[*] *Jacobs University Bremen gGmbH, D-28759 Bremen, Germany*
d.borrmann|a.nuechter@jacobs-university.de
[**] *University of Zagreb, Faculty of Electrical Engineering and Computing, Croatia*
marija.dakulovic|ivan.maurovic|ivan.petrovic@fer.hr
[***] *Faculty of Electrical Engineering, Sarajevo, Bosnia and Herzegovina* dinko.osmankovic|jasmin.velagic@etf.unsa.ba

Abstract: Heat and air conditioning losses in buildings and factories lead to a large amount of wasted energy. The Action Plan for Energy Efficiency of the Commission of the European Communities (2008) estimates that the largest cost-effective energy savings potential lies in residential (≈ 27%) and commercial (≈ 30%) buildings. Imagine a technology that creates a precise digital 3D model of heat distribution and heat flow enabling one to detect all sources of wasted energy and to modify buildings to reach these savings. This paper presents our overall approach to map indoor environments with thermal data in 3D.

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

1. INTRODUCTION

Recently a lot of work has been done to capture and reconstruct the world around us. Thermal imaging is state of the art in recording energy related issues of buildings. However the acquired images tell the user the precise temperature without the dimensions of the heat or air leak. Reliable solutions to 3D reconstruction based on images have not been presented, yet. Terrestrial laser scanning has been used for years to create 3D models. Registration algorithms from the geodesy and robotics community combine laser scan data acquired at different positions into complete models of the environment. Think of a technology that enables one to gage the environmental structure in 3D and thermal information at the same time. The availability of precise thermal 3D models will enable architects and construction engineers to inspect the model, to run simulations of heat and air flow and to use the gained information to modify existing buildings to reach the estimated energy savings. The goal of the project is to develop a fully automated system for thermal 3D mapping by a mobile robot. This paper presents our approach towards this goal.

Developing a fully automated system for thermal 3D mapping by a mobile robot is divided into three main parts: (i) advanced mutual calibration between the 3D sensor and the thermal camera, (ii) sensor placement planning, and (iii) 3D thermal surface modeling. Section 2 reviews the related work in the related research topics. Section 3 explains advanced mutual calibration between the 3D sensor and the thermal camera. Section 4 presents the sensor placement planning algorithm. Section 5 introduces our approach for 3D thermal surface model building. Experimental results are shown in section 6, and section 7 concludes the paper.

2. BACKGROUND AND STATE OF THE ART

To assess 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), if the focus lies on solving simultaneous localization and mapping (SLAM), i.e., on recovering the 3D structure of the environment and the sensors poses (position and orientation). Since reliable solutions to image based 3D reconstruction for thermal images have not been presented yet, we use the emerging technology of terrestrial laser scanning. 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.

[⋆] This work was supported by SEE-ERA.NET project ThermalMapper under the project number ERA 14/01. Project Webpage (2011)

Related work in inspection robotics includes work in human detection in thermal cameras using temperature signatures Markov and Birk (2007). Högner and Stilla (2007) present a modified van as surveying vehicle for acquiring thermal images in urban environments. However, in the focus are outdoor environments and image-based techniques like SFM. Prakash et al. (2006) present stereo imaging using thermal cameras, but focus on small scale applications. Iwaszczuk et al. (2011) suggest an approach to map terrrestrial and airborne infrared images onto existing building models. The model is textured by extracting polygonal parts from the image and mapping those onto the model using standardized masked correlation. Only a little work has been done for combining 3D scanners and thermal cameras. Cabrelles et al. (2009) present a methodology to exhaustively record data related to a World Heritage Monument using terrestrial laser scanning, close range photogrammetry and thermal imagery. 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 and use a total of eight natural control points with the help of the total station to relate the geometry between different sensors . Pelagottia et al. (2009) present a first automatic approach for multispectral texture mapping. Their method is based on the extraction of a depth map in the form of an image from the model geometry whose pixels maintain an exact correspondence with the vertices of the 3D model. Afterwards the registration with the chosen texture is done which is based on maximization of mutual information. 3D environment mapping using 3D scanners on mobile robots are subject of research Surmann et al. (2003); Nüchter (2009). Up to our knowledge modeling using 3D scanning and thermal imaging has not been done yet.

Sensor placement planning is needed for goal directed acquisition of 3D data. The task of the 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 about 3 minutes for one position, depending on 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 González-Baños and Latombe (2001). These methods are chategorized 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 autonomously navigate in an unknown environment and to build its 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 Ekman et al. (1997); Tovar et al. (2007). The majority of exploration algorithms is not reliable when applied under real conditions due to sensitivity to uncertainty of measurements, localization, and map building. A small divergence in localizing at the precomputed NBV point can lead to many unnecessary movements. Moorehead et al. (2001) include the uncertainty of the robot's pose into the exploration strategies. Recently, a number of sensor placement planning algorithms have been developed for 3D environment model reconstruction. Most methods take 3D scans based on a 2D exploration strategy Surmann et al. (2003). Blaer and Allen (2007) propose a 3D NBV method which plans additional viewing locations based on a voxel-based occupancy procedure for detecting holes in the model. Low and Lastra (2007) present a full non-model-based 3D NBV method based on exhaustive hierarchical 3D view metric evaluation. However, the computational complexity is still the major challenge to a practical NBV solution.

Another challenging task in building a 3D thermal model of an indoor environment is the reconstruction of the 3D mesh with added temperature scalar field to it. Wardlaw et al. (2010) evaluate the impact of 3D models onto the user's ability to interpret thermal information with the conclusion that a good representation of a 3D model can help the user, e.g., to locate heat sources or to detect thermal bridges. The point cloud representation is not the best way to present a virtual model of the environment. For visual perception, the best representation consists of a filled structure without holes, i.e., surfaces or meshes (Yan et al., 2008). As laser scanners are used in this paper, the gathered information is available as point clouds with added temperature information from the thermal imaging camera. The process of generating a surface from the set of points is called surface reconstruction. There are several methods to reconstruct the 3D surface from the point cloud. They can be divided into two distinct groups:

- Face-based reconstruction (triangulation)
- Iso-surface reconstruction (meshing)

The most common triangulation based algorithms are the Voronoi-based triangulation as presented by Amenta et al. (1998) and the triangulation-based Delaunay 3D algorithm as presented by Berg et al. (2008). Su and Drysdale (1996) present several implementations of the latter algorithm analyzing the performance. These algorithms require closed and organized geometry. This means the algorithms tend to close the structure of the model which is very good for digitization of cultural heritage sites as presented by Chrysanthou et al. (2011) and Altantsetseg et al. (2011). Iso-surface reconstruction, unlike the triangulation which reconstructs part-by-part, generates iso-surface, meaning that the model is generated by connecting the points with constant values, i.e., distance, pressure, temperature, etc.). Often used for iso-surface reconstruction is the Marching Cubes Algorithm (MCA) developed by Lorensen and Cline (1987). This algorithm starts with the space (containing the 3D scalar field or voxels) divided into cubes. After selecting a random cube and its eight neighbours, the algorithm determines the polygonal part of the iso-surface passing through the cube. There are modifications to the standard MCA presented by Chernyaev (1995) and Nielson and Hamann (1991) which deal with "ripples" caused by topological inconsistencies of the model (or point cloud acquisition). Since the space is divided equally it is very good 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 precisely calculate iso-values. Several methods are presented: ball pivoting by Bernardini et al. (1999), Poisson surface reconstruction by Kazhdan et al. (2006) and butterfly subdivision of surfaces by Mo-

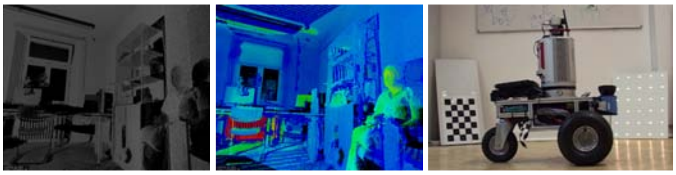

Fig. 1. Laser scan with reflectance (left) and thermal (middle) information. The robot Irma3D (right).

hamad and Bade (2009). These methods require very high resolution point clouds resulting in large data sets (millions of points) or small scale models, and are generally very slow. The problem of reconstruction of 3D model with added thermal field information will be addressed in detail in section 5.

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 Fig. 1). 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° C. It acquires images at a frame rate of 120 Hz and with an accuracy of 2° C with a field of view of approximately 40° × 64°. The laser scanner acquires data with a field of view of 100° × 360°. To achieve the full horizontal field of view the scanner head rotates around the vertical scanner axis when acquiring the datscanninga. We take advantage of this feature when acquiring image data. Since the camera is mounted on top of the scanner, it is also rotated. We acquire 9 thermal images during one scanning process to cover the full 360°. 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 fused with the thermal information. This processing consists of five steps that will be explained in this section.

Intrinsic Calibration of Thermal 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. A camera has unique parameters that define how a point (X, Y, Z) in world coordinates is projected onto the image plane. 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 and the known distance between those in world coordinates allows to formulate equations (1) and (2) as a linear least squares problem and solve for the calibration parameters.

For low resolution thermal cameras a chessboard pattern is more error-prone even after heating it with an infrared lamp. Instead a pattern with clearly defined heat sources such as tiny lightbulbs is suggested as it shows up nicely in thermal images thus enabling us to perform intrinsic calibration in the same way as for optical cameras. To detect the light bulbs in the thermal image 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 two big or too small. If the remaining number of regions is equal to the number of lightbulbs in the pattern the regions are sorted according to the pattern to allow for easy determination of correspondences.

Extrinsic Calibration – Thermal Camera and Laser Scanner. After calculating the internal parameters of the camera 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 they are unique to every camera view. Once all the points are in the camera coordinate system the projection to the image can be defined up to an factor s using equation (3) Bradski and Kaehler (2008):

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

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 than the maximum likelihood estimate of the transformation between the scanner and camera coordinate system is obtained by minimizing

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

where $\mathbf{A}$ is the intrinsic matrix, $\mathbf{R}_i$ the rotation matrix, $\mathbf{t}_i$ the translation vector, and $\mathbf{D}$ the distortion parameters. $\hat{\mathbf{p}}(\mathbf{A}, \mathbf{D}, \mathbf{R}_i, \mathbf{t}_i, \mathbf{P}_j)$ defines the projection of point $\mathbf{P}_j$ in image i, according to equation (3) and (2). 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, i.e., light bulbs arranged in a regular grid pattern as shown in the background of Fig. 1 (right). The position of the points on this patterns are known. Algorithm 1 detects the points in a laser scan.

3D to 2D Projection and Color Mapping. During the data acquisition phase laser scans and images are acquired

Algorithm 1 Calibration pattern detection in a laser scan.

```
Require: point cloud, specification of calibration pattern
discard points outside the area of the expected board
find the most prominent plane using RANSAC
project a generated plane model into the center of the detected plane
use ICP to fit the plane model to the data points
return lightbulb positions according to ICP result
```

simultaneously. After determining the relations between scanner and camera in the calibration step this relation is used directly to assign temperature values to the points.

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 jump edges. Consequently small calibration inaccuracies have a large impact on the results. To solve this problem we take advantage of the fact that if a point belongs to an object there will be more points on that object. We take all points that are projected onto one pixel and its neighboring pixels. The points are clustered depending on their distance to the scanner. A heuristic based on distance to the 3D scanner and size of the cluster determines effectively which points are considered and enhanced with thermal information. This removes also some correct thermal information but the improvement prevails.

Scan Registration. Laser scans acquired at different positions are registered into one common coordinate system using 6D SLAM from *The 3D Toolkit* (3DTK) Andreas Nüchter et al. (2011). It estimates high-precisely the scanner pose with 6 degrees of freedom.

4. SENSOR PLACEMENT PLANNING

The sensor placement planning algorithm is based on our previous approach presented in Đakulović et al. (2011). The inputs are range values in $n = 720$ uniformly distributed directions from the 360° field of view extracted from the 3D laser scanner such that all range data lie in the plane parallel to the floor plane. It is assumed that the environment model is unknown and is incrementally built at each scanning position. The model is represented at three abstraction levels. At the lowest level the occupancy grid map is used for storing 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 real environment edges 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 searched to find the best goal position for the path planning module. We assume a setup with a resolved robot localization problem (the GMAPPING module under ROS[1] is used). 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 method described in Seder and Petrović (2007). The sensor placement algorithm is composed of three steps which are executed at each scanning position: (1) vectorization – extracting lines from range data, (2) creation of the exploration polygon – building the most recent model of the environment, and (3) selection of the next sensor position – searching the next goal for the path planning module. These three steps are explained in the following.

Vectorization. The main goal of vectorization is grouping of the range data $\mathcal{R} = \{r_k \mid k = 1, \ldots, n\}$ into M subsets, where M is not known a priori. Each subset contains almost collinear points within some error limit. For each subset the line segment is obtained by a least squares method. First, the Progressive Probabilistic Hough Transform PPHT (from the OpenCV library Matas et al. (1998)) is carried out over the range data. PPHT calculates initial estimations of line segments which are then used to group all range data around the calculated line segments according to their distances $\delta\rho_k$ from the lines. Lines are represented in polar coordinates $L = [\rho \ \ \alpha]^T$, where ρ defines the normal distance to the origin and α defines the angle between normal and positive x-axis. For every measurement point the distance $\delta\rho_k$ is obtained as

$$\delta\rho_k = |d_k \cos(\alpha - \Theta_k) - \rho|, \tag{5}$$

where $r_k = [d_k \ \ \Theta_k]^T$ refers to k-th laser point in polar coordinates. If the distance $\delta\rho_k$ is below some treshold value $\Delta\rho$ the point belongs to set $\mathcal{R}_p$ for the corresponding line segment L_p.

The next step calculates more precise line parameters by a least squares line fitting algorithm which solves the following minimization problem

$$L = \underset{\alpha,\rho}{\operatorname{argmin}} \sum_{k=1}^{N} \delta\rho_k^2, \tag{6}$$

where N is the number of points in each subset of range data. Line segments are deterimined by trimming the given line at extreme endpoints from the subset of range data. At the end of the vectorization process real environment edges (walls, obstacles) are represented by line segments.

Creation of Exploration Polygon. For selecting the next sensor position, for gridmap creation, and path planning a polygonal representation of the environment is used. At each scanning position p_i two polygons are calculated: the measurement polygon P_i created from the vectorization of the newest range data, and the exploration polygon EP_i created from the exploration polygon from the previous scanning position EP_{i-1} and the new measurement polygon P_i. Initially the exploration polygon is empty, $EP_0 = \emptyset$. Both polygons are composed of real edges and so-called jump edges. The real edges are detected in the

[1] Robot Operating System, http://www.ros.org

vectorization step and present line segments which are found in the environment, e.g., walls, obstacles, while the jump edges are connections between two real edges and are used as separation between explored and unexplored regions.

Calculated line segments from the vectorization step create the measurement polygon as follows. The line segments are sorted such that their ending points have increasing angles when transformed into the polar coordinate system from $[-\pi, \pi]$. Connecting the ending points $s_1, \ldots, s_n$ defines the polygon P_i at scanning position p_i. Now, the polygon P_i 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 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 Vatti (1992)). In each step, EP_i is updated as $EP_i = EP_i \bigcup P_i$. The union of two polygons maintains only those edges most distant from the robot's point of view in the polygon from both polygons, and thus ensures that new jump edges are not created within previously explored regions. This property makes the planner also robust to small localization errors. Jump edges that are larger than the preset value Δr are considered for the selection of the next sensor position. The minimal length of the jump edge Δr is chosen in accordance to the robot dimensions and enables discarding too small jump edges (i.e., robot cannot pass through too narrow passages). If the nonempty extended exploration polygon EP_i contains no jump edges, then it is considered as the reliable polygonal description.

Selection of the Next Sensor Position. We use a simple heuristic criterion for selecting the next sensor position similar to Ekman et al. (1997). By taking a scan in front of the jump edge it is easy to imagine that we will gain greater amount of the new information than when scanning further from the jump edge inside of the explored area. Therefore, each jump edge is assigned one candidate scanning position. 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 robot's dimensions 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 according to the maximization of a criterion that estimates the amount of unexplored regions seen from each potential position.

The measure for the amount of unexplored region possibly seen from a candidate position (denoted by p_k) is defined by the angle in the triangle that is defined by the k-th jump edge and the k-th candidate position (Fig. 2). Other angles need to be taken into account because it is possible to see other jump edges from one scan position. The criterion used is as follows:

$$I_j = k_1 \frac{1}{d_j} + k_2 \sum_{i=1}^{N} \alpha_{ij}, \tag{7}$$

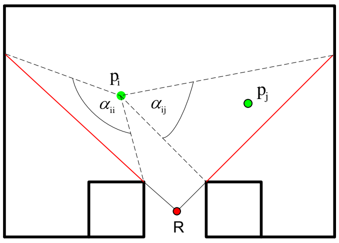

Fig. 2. Selection criterion based on angles of visibility.

where d_j is the length of the shortest path from the path planning module between the current robot position and the j-th candidate position, 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. The parameters are set in order to treat both terms in (7) equally. 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. The whole sensor placement algorithm is summarized in Algorithm 2.

Algorithm 2 Sensor placement planning

```
Require: 2D range data, exploration polygon from previous scans
vectorization
sort detected lines by angle and create measurement polygon
find jump edges connecting adjacent real edges
discard jump edges smaller than minimal passage width
obtain union of measurement polygon and exploration polygon from previous scans
discard jump edges within exploration polygon
calculate potential measurement positions in front of jump edges
choose the best position according to criterion
return the next sensor position
```

5. 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 Fig. 1). Switching between the different views enables the user to detect sources of wasted energy and to locate them clearly in the 3D view. All 3D points have a guaranteed accuracy of 1-5 centimetres. However, the point cloud representation is not the best way to present the virtual model of an environment. For human eyes, the best representation consists of a filled structure without holes, i.e., surfaces, meshes (Yan et al., 2008). As presented in previous sections, the gathered information are represented as point clouds with added temperature information from a thermal imaging camera. The process of generating surfaces from the set of points is called surface reconstruction. In this section surface reconstruction with added temperature field information is addressed. Through the next subsections the problematics of the MCA based approach is discussed. This includes the estimation of iso-values, which is a very important step before the actual MCA-based iso-surface reconstruction and mesh generation. Finally, the temperature field mapping onto the newly reconstructed model of the indoor environment is presented.

5.1 Estimation of iso-values.

To generate the iso-surface more quickly (although not very precisely) a method different to the beforementioned ones is used. One of the best estimation mechanism is a Gaussian filter. This is a probabilistic method that attempts to estimate the iso-value of each point in the data set by injecting the input point into stuctured geometry Schroeder et al. (2006)). This is accomplished by assigning a Gaussian distribution to every input point, and then aggregating the final iso-surface by applying some aggregation operator, i.e., minumum, maximum, or sum.

Given the point $\mathbf{p}$, the Gaussian distribution function $F(\mathbf{p})$ is calculated as:

$$F(\mathbf{p}) = s \cdot \exp\left[\cdot \left(\frac{r}{R}\right)^2\right] \tag{8}$$

where:
$\mathbf{p}$ - current voxel or sample point (x, y, z),
s - scale factor, can be multiplied by scalar value,
f - decay factor,
r - distance $|\mathbf{p}_r - \mathbf{p}|$ where $\mathbf{p}_r$ is a voxel in the neighbourhood of $\mathbf{p}$,
R - radius of influence.

By changing the listed parameters of the Gaussian distribution function it is possible to change the width of the bell, decaying the influence on the neighbouring points/voxels. This method can improve the results if normals in every point are provided.

5.2 Applying the Marching Cubes Algorithm.

The next procedure presented here is iso-surface reconstruction by using MCA, a well-known algorithm for calculating the surface from iso-values estimated by the procedure described in 5.1. It is commonly used in medical volumetric data representation of MRI and CT images Cline et al. (1991); Motiur et al. (2010). The idea of the algorithm is explained in Algorithm 3 using the 2D variant, i.e., Marching Squares.

This is easily extended to the 3D case by replacing lines with surfaces, and squares with cells (voxels), respectively.

5.3 Temperature Scalar Field Mapping

Gaussian estimation of iso-values and subsequent MCA reconstruction do not deal with scalar values of temperature acquired by the thermal imaging camera. These values have to be mapped onto the reconstructed surface. One problem that emerges is that it is not possible to establish one-to-one correspondences between the point cloud and the reconstructed model, i.e., a point in the point cloud is not the same point in the reconstructed model. To overcome this problem, foremost, the k-D tree of the model is constructed, so every point is organized into a tree. This enables fast searching through the model, e.g., range searches and nearest neighbour searches. Now, it is possible to find corresponding point in the reconstructed model to every point in the point cloud. After finding such a point, it gets the scalar value from its corresponding point in the point cloud, since MCA produces the model that is as close as possible to the original point cloud. The pseudo code for this procedure is given in Algorithm 4.

Algorithm 3 Marching Squares pseudo-code.

```
Require: Structured data set with calculated iso-values
for every point in the grid do
    set the status of the point to inside or outside the iso-line
end for
for every square in the grid do
    for every line segment that has a point outside and a point inside the figure do
        Add a point in the list t placed in the middle between the two points
    end for
end for
Draw sequential lines between the points found in the list t.
```

Algorithm 4 Scalar Field Mapping pseudo code.

```
Require: Reconstructed iso-surface and point cloud with scalars, original point cloud
Build a $k$-D tree for the reconstructed iso-surface model
for every point $p$ in the point cloud do
    Find the nearest point $\mathbf{p}_m$ in the model to point $\mathbf{p}$
    Assign the scalar value of the point $\mathbf{p}$ in the point cloud to point $\mathbf{p}_m$ from the model
end for
```

6. RESULTS

Experiments were carried out in a research building at Jacobs University Bremen, Germany. Fig. 3 shows the trajectory of the robot while exploring the unknown environment together with the generated polygonal model of the environment. The 18 scanning positions are marked with circles. The sequence of snapshots for the first five

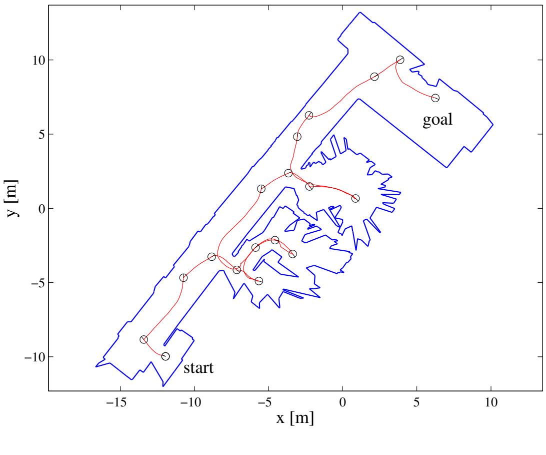

Fig. 3. Robot trajectory and scan positions while exploring the environment with obtained polygonal model.

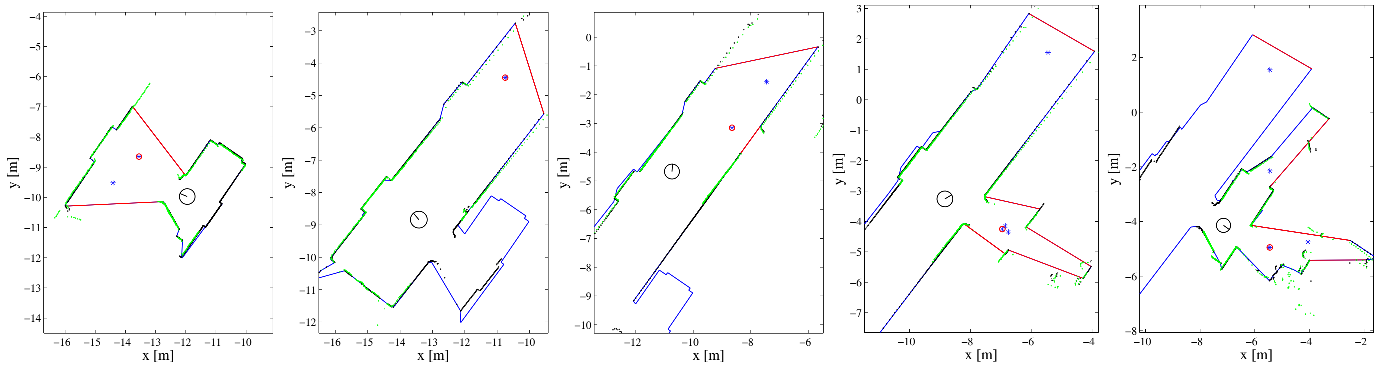

Fig. 4. A sequence of snapshots for the first five scanning positions for the sensor placement planning algorithm.

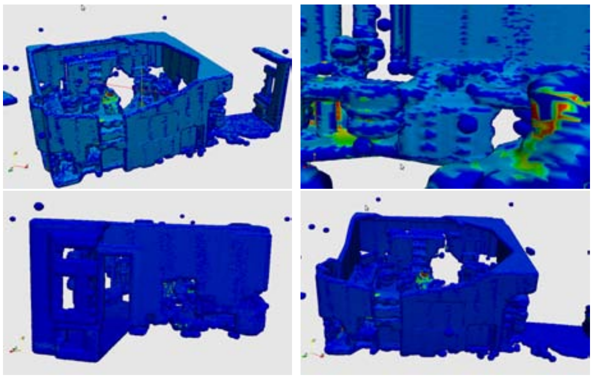

Fig. 5. Reconstruction from scanned indoor environment

scanning positions are shown in Fig. 4. Jump edges are colored red, the exploration polygon is colored blue, candidate scanning positions are noted by asterisks, while the chosen next position is marked by a red circle. Range readings from the 3D laser scanner are noted by black dots, range readings from the 2D laser scanner which are used for obstacle avoidance between two scanning positions are noted by green dots, respectively.

We also present first results of the reconstruction of the 3D thermal model. Fig. 5 shows the reconstruction of one part of the scanned indoor environment, a scan taken in a laboratory. The model contains close to 300,000 points for a volume of less than $20m^3$. Compared to the data sets presented in the papers by Bernardini et al. (1999), Kazhdan et al. (2006) and Mohamad and Bade (2009) this data set is rather small. Therefore, the precision of the reconstruction leaves room for improvement. 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, larger space subdivision or point cloud density is needed. This comes at the expense of larger computing power and memory requirements to store the model.

To evaluate the obtained model visual perception of an expert is necessary. The presented methodology attempts to minimize the local error of the reconstruction and does not deal with overall precision. To further assess the precision the generated mesh can be directly compared to the original point cloud by discretizing it back into a point cloud. In Fig. 6 the precision can be evaluated visually. Up to our knowledge there exists no method for computational evaluation of the precision for the reconstructed model.

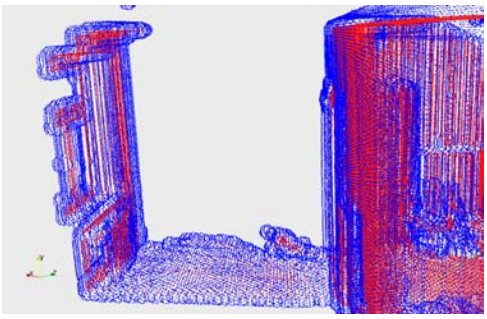

Fig. 6. Comparison of the original point cloud (red) and the point cloud of the reconstructed mesh (blue)

7. CONCLUSIONS

This paper presents a complete system for autonomous exploration and thermal mapping of an indoor environment. To improve the usability of the robotic system further steps are necessary. We plan to detect points of interest in the thermal model, i.e., heat leaks by analyzing the gradient of temperature values in the model. Furthermore, an additional optical camera will be included into the setup. With the added information a human expert can easily evaluate the points of interest and possibly identify the heat sources immediately, which will be further used for automation using machine learning. Future work will extend the exploration algorithm described above with full 3D information collected from the 3D scan. We are developing a model based algorithm with voxel representation of the environment. To reduce the computational complexity caused by the huge number of voxels needed to model the whole environment we are developing a room extraction algorithm which will enable voxel representation and 3D sensor placement planning to be applied sequentially room by room.

When a room is extracted and all voxels inside it are captured or marked as non-capturable, the sensor placement planning switches back to 2D exploration until the next room is detected. Therefore, our future work focuses on room extraction and voxel based sensor placement planning in 3D.

REFERENCES

Altantsetseg, E., Muraki, Y., Chiba, F., and Konno, K. (2011). Surface reconstruction of stone tools by using four-directional measurement machine. *The International Journal of Virtual Reality.*

Amenta, N., Bern, M., and Kamvysselis, M. (1998). A new voronoi–based surface reconstruction algorithm.

Andreas Nüchter et al. (2011). 3DTK – The 3D Toolkit. http://slam6d.sourceforge.net/.

Berg, M.d., Cheong, O., Kreveld, M.v., and Overmars, M. (2008). *Comp. Geometry: Algorithms and Applications*. Springer TELOS, Santa Clara, USA, 3rd ed.

Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., and Taubin, G. (1999). The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5, 349–359.

Blaer, P. and Allen, P. (2007). Data acquisition and view planning for 3-D modeling tasks. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 417–422. IEEE.

Bradski, G. and Kaehler, A. (2008). *Learning OpenCV, Computer Vision with OpenCV library*. O'Reilly Media.

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

Chernyaev, E.V. (1995). Marching cubes 33: Construction of topologically correct isosurfaces. Technical report.

Chrysanthou, Y., Cain, K., Silberman, N., (editors, F.N., Bendels, G.H., Degener, P., Körtgen, M., and Klein, R. (2011). Image-based registration of 3D-range data using feature surface elements.

Cline, H.E., Lorensen, W.E., Souza, S.P., Jolesz, F.A., Kikinis, R., Gerig, G., and Kennedy, T.E. (1991). 3D surface rendered mr images of the brain and its vasculature. *J. Of Comp. Assisted Tomography*, 15(2), 344–351.

Comm. of the Europ. Communities (2008). Addressing the challenge of energy efficiency through Information and Communication Technologies. Communications to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, COM(2008) 241 final.

Đakulović, M., Ileš, S., and Petrović, I. (2011). Exploration and Mapping of Unknown Polygonal Environments Based on Uncertain Range Data. *AUTOMATIKA: J. Contr., Meas., Electr., Comp., Comm.*, 52(2), 118–131.

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

González-Baños, H. and Latombe, J. (2001). A randomized art gallery algorithm for sensor placement. *SCG'01: Proc. of the 17th Ann. Symp. on Comp. Geom.*, 232–240.

Högner, L. and Stilla, U. (2007). Texture extraction for building models from ir sequences of urban areas. In *Proc. of URBAN/URS 2007*.

Iwaszczuk, D., Hoegner, L., and Stilla, U. (2011). Matching of 3D building models with IR images for texture extraction. In *JURSE 2011*. Munich, Germany.

Kazhdan, M., Bolitho, M., and Hoppe, H. (2006). Poisson surface reconstruction. In *Proc. of 4th Eurographics Symp. on Geom. Processing*, SGP '06, 61–70. Eurographics Association, Aire-la-Ville, Switzerland.

Lorensen, W.E. and Cline, H.E. (1987). Marching cubes: A high resolution 3D surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21, 163–169.

Low, K. and Lastra, A. (2007). Efficient constraint evaluation algorithms for hierarchical next-best-view planning. In *Proc of 3DPVT 2006*, 830–837.

Markov, S. and Birk, A. (2007). Autonomous victim detection in rescue operations by modelbased reproduction of 3D scenes from 2D thermal images. In *KI 2007: Advances in Art. Intell., Proc. of 30th Ann. German Conf. on AI (KI '07)*, Osnabrück, Germany.

Matas, J., Galambos, C., and Kittler, J. (1998). Progressive probabilistic hough transform. *Proc. British Machine Vision Conference BMVC98*, 256–265.

Mohamad, I. and Bade, A. (2009). Butterfly subdivision scheme used for the unorganized points reconstruction in virtual environment. In *Proc. of 2009 Inter. Conf. on Comp. Techn. and Develop. - Vol 02*, ICCTD '09, 345–349. IEEE Computer Society, Washington, DC, USA.

Moorehead, S.J., Simmons, R., and Whittaker, W.L. (2001). Autonomous exploration using multiple sources of information. In *Proc. of ICRA '01*. Seoul, Korea.

Motiur, R., Shorif, U., and Mosaddik, H. (2010). 3D segmentation and visualization of left coronary arteries of heart using ct images. *J. Comp. Appl.*, (2), 88–92.

Nielson, G.M. and Hamann, B. (1991). The asymptotic decider: resolving the ambiguity in marching cubes. In *Proc. of the 2nd Conf. on Visual. '91*, VIS '91, 83–91. IEEE Computer Society Press, Los Alamitos, CA, USA.

Nüchter, A. (2009). *3D Robotic Mapping: The Simultaneous Localization and Mapping Problem with Six Degrees of Freedom*. No. 52 in Springer Tracts in Adv. Robotics.

Pelagottia, A., Mastio, A.D., Uccheddu, F., and Remondino, F. (2009). Automated multispectral texture mapping of 3D models. In *EUSIPCO 2009*. Glasgow, Scotland.

Prakash, S., Pei, L.Y., and Caelli, T. (2006). 3D mapping of surface temperature using thermal stereo. In *Proc. of ICARCV '06*.

Project Webpage (2011). Project ThermalMapper. http://www.faculty.jacobs-university.de/anuechter/thermalmapper.html.

Schroeder, W., Martin, K., and Lorensen, B. (2006). The visualization toolkit: an object-oriented approach to 3D graphics.

Seder, M. and Petrović, I. (2007). Dynamic window based approach to mobile robot motion control in the presence of moving obstacles. In *Proc. of ICRA '07*, 1986–1991.

Su, P. and Drysdale, R.L.S. (1996). A comparison of sequential delaunay triangulation algorithms.

Surmann, H., Nüchter, A., and Hertzberg, J. (2003). An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *J. RAS*, 45(3–4), 181–198.

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

Vatti, B. (1992). A generic solution to polygon clipping. *Comm. of the ACM*, 35(7), 56–63.

Wardlaw, J., Gryka, M., Wanner, F., Brostow, G., and Kautz, J. (2010). A new approach to thermal imaging visualisation. EngD Group Project, Univ. Coll. London.

Yan, Z., Hu, L., Chen, H., and Lu, F. (2008). Computer vision syndrome: A widely spreading but largely unknown epidemic among computer users. *Comput. Hum. Behav.*, 24(5), 2026–2042.