A Study of Projections for Key Point Based Registration of Panoramic Terrestrial 3D Laser Scans*

HamidReza Houshiar, Jan Elseberg, Dorit Borrmann, Andreas Nüchter

Abstract

This paper surveys state of the art image features and descriptors for the task of 3D scan registration based on panoramic reflectance images. As modern terrestrial laser scanners digitize their environment in a spherical way, the sphere has to be projected to a two-dimensional image. To this end, we evaluate the equirectangular, the cylindrical, the Mercator, the rectilinear, the Pannini, the stereographic, and the z-axis projection. We show that the Mercator and the Pannini projection outperform the other projection methods.

1 Introduction

Laser scanners are state of the art in modelling architectural structures, historical sites and even entire cities or landscapes. Digitizing environments without occlusions requires multiple 3D scans, i.e., 3D point clouds. To create a correct and consistent model, the scans have to be merged into one coordinate system. This process is called registration. A popular algorithm to automatically merge two independently acquired 3D point clouds was already presented in 1991. This algorithm is called the iterative closest point algorithm (ICP) [8]. ICP does not rely on features, instead it correlates points with the closest Euclidean distance. To

be successful, it needs a good starting estimate. Feature based methods establish correspondences by comparing descriptions of points of interest, i.e., features. Fully automatic feature based registration procedures require neither manual feature selection, manual establishing of correspondences, nor the placement of artificial landmarks.

Existing automatic methods use either structural 3D features or image features. The latter are obtained from images generated from 3D scans. In addition to measuring distances, current laser scanners are able to determine the amount of light returning to the scanner. This reflectance information of the scanned surface is used to create a gray scale image.

2D laser scanners use a rotating mirror to deflect the emitted laser beam and measure the distance r to an object under a certain angle φ. Based on the orientation of the mirror a polar representation (φ, r) of a surface profile is determined which can be transformed into Cartesian coordinates. 3D laser scanners either use a second mirror or rotate a 2D laser scanner to generate 3D point clouds of the scanned environment. The resulting 3D data in spherical coordinates (θ, φ, r), with θ as the angle of the second rotation, can again be transformed into Cartesian coordinates (x, y, z). Figure 1 shows an illustration of a scanning process.

This paper studies feature based registration methods. We evaluate various feature detectors and feature description methods that were previously used for camera images with respect to their applicability for registering scans. While the SIFT features have previously been used for scan matching [9], we also employ other descriptors and detectors and do an experimental evaluation using several data sets obtained by a terrestrial 3D laser scanner. We focus on panoramic scans, i.e., full 360° scans. Additionally, we put emphasis on the projection methods for the 3D scans. Up to our knowledge, the issue of projecting the spherical scanner data to a 2D image and the related effects on scan registration have not been researched yet.
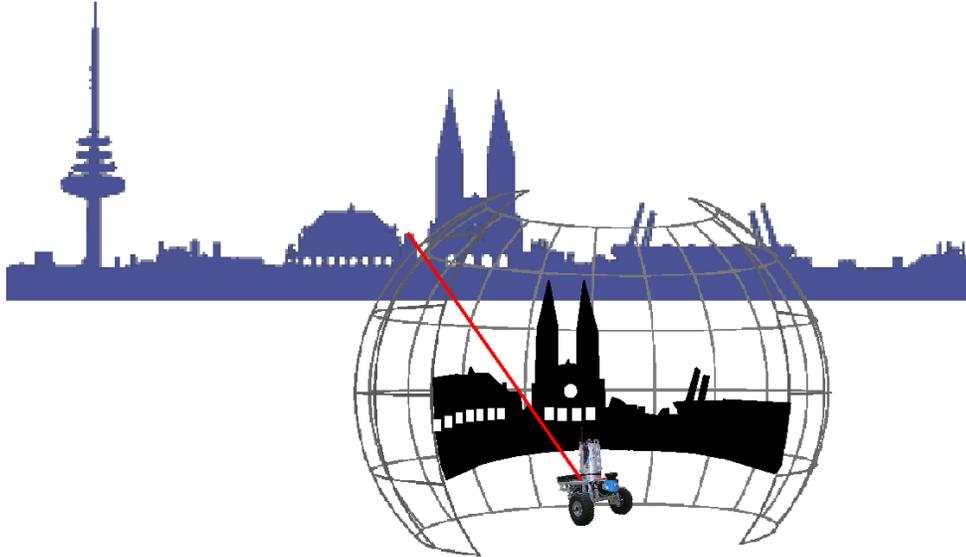
Figure 1: A Riegl VZ-400 3D laser scanner with a field of view of $100° \times 360°$ mounted on the mobile robot Irma3D is used to scan the city center of Bremen.

2 State of the art

2.1 Automatic pairwise scan-to-scan and photo-to-scan registration

Many state of the art registration methods rely on initial pose (position and orientation) estimates, acquired by global positioning systems (GPS) or local positioning using artificial landmarks or markers as reference [48]. Pose information is hard to acquire and in many scenarios prone to errors or not available at all. Thus, registration without initial pose estimates and place recognition are highly active fields of research. Hansen et al. consider an application of scale-invariant feature detection using scale-space analysis suitable for use with wide field of view cameras. They map the image to the sphere and obtain scale-pace images as the solution to the heat (diffusion) equation on the sphere [23, 24]. Lee et al. presents a matching method in order to find the correspondences of

features in two omnidirectional images. Dominant corresponding feature pairs are found using a proximity matrix and a sum of squared differences (SSD) based similarity matrix, and then the remaining feature matching is accomplished by dynamic time warping (DTW) [30]. Parida et al. present a unified approach to detect, classify, and reconstruct junctions in images. Their approach is a combination of the two paradigms: edge detection followed by grouping of edges to form junctions (via Dynamic Programming) and treating a junction as a template matching process. They use a template deformation framework using the minimum description length (MDL) principle[37]. By extending this approach, Försnter et al. proposed a novel method for detecting scale invariant keypoints. a scale space mechanism for junction type features[22]. Common appearance based place recognition approaches often rely only on camera data and are not suitable for laser scans [10, 15, 16, 28, 47].

Aside from range values laser scanners record the intensity of the reflected light. These intensities provide additional information for the registration process. Böhm and Becker suggest to use SIFT features for automatic registration and present an example of a successful registration on a 3D scan with a small field of view [9]. Wang and Brenner extended this work by using additional geometry features to reduce the number of matching outliers in panoramic outdoor laser scans [49]. Kang et al. propose a similar technique for indoor and outdoor environments [27]. Weinmann et al. use a method that is based on both reflectance images and range information. After extraction of characteristic 2D points based on SIFT features, theses points are projected into 3D space by using interpolated range information. For a new scan combining the 3D points with 2D observations on a virtual plane yields 3D-to-2D correspondences from which the coarse transformation parameters can be estimated via a RANSAC based registration scheme including a single step outlier removal for checking consistency [51]. They expend their method in [50] to calculate the order of the scans in unorganized terrestrial laser scan data by checking the similarity of the respective reflectance images via the total number of SIFT correspondences between them. Bendels et al. exploit intensity images often recorded with the range data and propose a fully automatic registration technique using 2D-image features. The fine registration

of two range images is performed by first aligning the feature points themselves, followed by a so-called constrained-domain alignment step. In the latter, rather than feature points, they consider feature surface elements. Instead of using a single 3D-pooint as feature, they use the set of all points corresponding to the image area determined by the position and scale of the feature [7].

Other approaches rely only on the 3D structure. Brenner et al. use 3D planar patches and the normal distribution transform (NDT) on several 2D scan slices for a coarse registration [12]. Similarly, Pathak et al. evaluate the use of planar patches and found that it is mostly usable [38]. A solution using the NDT in 3D is given in [33]. While this approach computes global features of the scan, several researchers use features that describe small regions of the scan for place recognition and registration [26, 46, 3]. Flint et al. use a key point detector called THRIFT, to detect repeated 3D structures in range data of building facades [20].

In addition to coarse registration, many authors use the well-known iterative closest point algorithm (ICP) for fine registration [8, 53, 14]. ICP requires no computation of features. Instead, it matches raw point clouds by selecting point correspondences on the basis of smallest distances and by minimizing the resulting Euclidean error. This iterative algorithm converges to a local minimum. Good starting estimates improve the matching results drastically, i.e., they ensure that ICP converges to a correct minimum.

Another field of research is the co-registration of point clouds and camera data. Stamos and Allen present a semi-automatic method for image to model registration of urban scenes, where 3D lines are extracted from the point clouds of buildings and matched against edges extracted from the images [45]. Aguilera et al. use the Förstner operator [21] to exploit its high accuracy and reliability in the localization of the interest points, and its robustness to noise [2]. Meierhold et al. use SIFT features for the task [34]. Böhm et al. also present a successful point cloud to camera matching algorithm [9]. Wu et al. use a novel viewpoint independent patches (VIP) on image sequence of urban scenes. They use structure from motion to compute the depth map and camera positions. VIP's are features that can be extracted from textured 3D models which combine images with corresponding depth maps. a single VIP correspondence uniquely defines

the 3D similarity transformation between two scenes [52]. Köser and Koch extend the 2D feature concept into third dimension by producing a descriptor for a constructed 3D surface region. They combine the concepts of stable texture interest points and 3D geometry. Their feature represents texture and surface information in a perspective invariant fashion by computing a normal view onto the surface. Their approach can be exploited for structure from motion, for stereo cameras or alignment of large scale reconstruction [29].

2.2 Image features and feature descriptors

In previous work, many algorithms for extracting image features, so-called key points, have been proposed. Descriptors are used to encode the features and in the automatic matching phase, these descriptors are compared. Next, we describe the features and descriptors used in this study. There are two types of features in this study. First group that contains SIFT and SURF is more robust and more invariance to noise, rotation and camera position. The second group has less resource consumption and has optimized for real-time performance which, contains FAST, ORB and CenSurE.

2.2.1 Scale Invariant Feature Transform (SIFT)

The Scale Invariant Feature Transform is a popular algorithm in computer vision to describe and determine distinctive invariant local features from images that can be used to perform reliable matching between different views of an object or scene. It is important that extracted features are invariant to scale, rotation and provide robust matching across a substantial range of affine distortions, change in 3D viewpoint, addition of noise and change in illumination. SIFT features fulfil all these requirements. A high number of features are found using an efficient algorithm. As SIFT features are distinctive, a single feature is likely to be correctly matched to a large database of features, providing a basis, e.g., for object recognition. The major steps of the SIFT feature extraction process are scale-space extrema detection, key point localization, orientation assignment, and key point description.

The first step of feature detection is to identify interest points repeatedly under differing views and scales. Initially an image pyramid is built by applying Gaussian filters of different sizes to the image. This allows for a search for the features over all scales and image locations. Subsequently, differences between consecutive levels of Gaussian convolved images are computed for efficient detection of stable feature location in scale space. This step is referred to as Difference of Gaussians. Afterwards local maxima of the Difference of Gaussians are computed by comparing each sample point to its eight neighbours in the current image and nine neighbours in the scale above and below. All local maxima that do not correspond to edges in the image and that have high contrast are selected as features.

The process continues by determining the orientation of features based on local image properties. Invariance to image rotation is achieved by representing the feature descriptor relative to the orientation of the feature. This is fulfilled by calculating magnitudes $m(x,y)$ and orientations $\theta(x,y)$ of the local neighbourhood around a feature in the image L of the scale where the feature was detected.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y+1))^2} \quad \text{(1a)}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right) \quad \text{(1b)}$$

Afterwards an orientation histogram is created. The magnitude of each neighbourhood point, weighted by a Gaussian-weighted circular window is added to the bin as determined by the angle $\theta(x,y)$. The largest peak is selected as the orientation of the feature. The special case of multiple substantial peaks is treated as separate features with same position and different orientation.

The next step computes the descriptor by considering the surrounding area of the feature. In a window of size 16×16 pixels, oriented according to the feature orientation, the orientation and magnitude for each pixel are computed. The window is grouped into a 4×4 array. In each of the 4 pixel wide cells an orientation histograms with 8 bins is computed in the same manner as in the previous step. A feature descriptor of $4 \times 4 \times 8 = 128$ dimensions is constructed by concatenating the orientation histograms.

The SIFT feature matching process utilizes a modified version of the k-d tree search known as Best Bin First [6] due to inefficiency of the standard k-d tree for high dimensional spaces. The ratio of the Euclidean distance between the feature and the nearest neighbour to the feature and the second nearest neighbour is proposed to determine a positive match in the matching process. This method is utilized in favour of the simpler method of thresholding the distance, considering that the determination of certain thresholds is a demanding task. [31, 32]

2.2.2 Speed-Up Robust Features (SURF)

The SURF algorithm is a relatively new scale- and rotation- invariant detector and descriptor[5, 4]. The SURF features encompass a satisfactory repeatability, distinctiveness, and robustness, yet they can be computed and matched fast. SURF relies on integral images for image convolution. It is based on existing methods which use a Hessian matrix-based measure for the detector and a distribution-based descriptor. The search for distinctive image points is divided into two steps: First, interest points are selected in distinguishable and repeatedly detectable locations in the image. Next, the descriptor of the detected point describes the intensity of its neighbourhood. The descriptor of the interest point is distinctive and robust to noise.

The SURF interest point detection is based on Hessian detectors. SURF employs image filters that are simple approximations of the second order derivative of the Gaussian. The filters are so simple that the convolution can be computed extremely fast by the help of integral images. The filters vary in size, starting with 9×9 pixels at the finest resolution. Using differently sized filters results in a similar situation to the image pyramid in SIFT. Local maxima are detected in the image pyramid by non-maximum suppression in a $3 \times 3 \times 3$ neighbourhood. Maxima below a threshold are eliminated. The remaining points are the detected features. In this process scale invariance is achieved by up-scaling the filter rather than reducing the image size.

The SURF descriptor represents the distribution of the intensity within the neighbourhood of the interest point. The descriptor is a vector of dimension

64 and is based on the distribution of the first order Haar wavelet responses in the x and y direction. Again integral images are utilized for speed. The SURF descriptor generation contains two steps: First, determine a reproducible orientation based on information from a circular region around the interest point to make the SURF descriptors invariant to rotation. Second, construct a square region aligned to the assigned orientation and extract the descriptor.

The dominating orientation is estimated with a sliding orientation window. The Haar wavelet responses of the points surrounding the feature are 2D orientation vectors. For each orientation window the Haar wavelet responses falling inside the window are summed up. The window with the largest orientation vector is selected as the orientation of the interest point. Afterwards a square region around the interest point is constructed and oriented along the determined orientation. This region is divided into 4×4 square sub-regions. For each sub-region the Haar wavelet responses at 5×5 regularly arranged sample points are computed. The wavelet responses are summed up in each sub-region. In order to include the information about the polarity of the intensity changes, the sum of absolute values of the response are calculated. Therefore, each sub-region has a four-dimensional descriptor vector for its intensity structure. Concatenating these vectors for all 4×4 sub-regions results in a 64 dimensional descriptor vector.

2.2.3   Features from Accelerated Segment Test (FAST)

The majority of feature detection algorithms compute corner response functions across the image. An edge in an image is the boundary between two regions of an image which defines a change in intensity. However, the direction of a boundary changes rapidly at corners. The FAST feature detector determines its features by examining a small circle with a boundary of sixteen pixels around corner candidates [39, 40]. A candidate p is considered a feature if the intensity of a set of n contiguous pixels in the circle are all brighter or darker then the intensity of the candidate pixel by some threshold. n was chosen to be twelve by [39]. The testing process is optimized by examining pixels at the four compass

directions first to reject candidate pixels more rapidly. A pixel is a feature if three of the corner points are darker or brighter than p by the threshold. The full segment test is applied to the remaining candidates from the optimization process by examining all sixteen pixels in the circle. This is a sufficiently fast detector and shows high performance. However, the optimization does not generalize for $n < 12$ and the knowledge from the first 4 steps is discarded and also multiple features are detected adjacent to one another. Rosten and Drummond present a machine learning algorithm to address the first two weaknesses of the original FAST feature detector and a non-maximal suppression has been utilized for the latter weakness.

2.2.4 Binary Robust Independent Elementary Features (BRIEF)

BRIEF is an efficient feature descriptor that utilizes binary strings [13]. Binary strings are highly distinguishable even with relatively few bits. Descriptors are directly computed from image patches. The individual bits are acquired by comparing intensities of pairs of pixels. Hamming distances are utilized for the matching process to evaluate the similarity between descriptors. This is more efficient in comparison to the more common L_2 norm. BRIEF shows that only 256 bits, or even 128 bits, often suffice to obtain superior matching results. Therefore, BRIEF is very fast in both the process of constructing and matching.

BRIEF is based on the classification of image patches by the pairwise intensity comparison of a relatively small number of pixel pairs. Each bit τ in the descriptor corresponds to one pair of pixels X, Y in the smoothed image. The bits are set as follows:

$$\tau(X,Y) := \begin{cases} 1 & \text{if } I(X) < I(Y) \\ 0 & \text{otherwise.} \end{cases}, \tag{2}$$

where $I(X)$ is the intensity of the pixel X. The descriptor is defined by n pairs (X, Y), and the bit string presentation is given by:

$$f_n := \sum_{1 \leq i \leq n} 2^{i-1} \tau(X,Y). \tag{3}$$

The pairs of pixels to compare are randomly selected. Different types of probability distributions can be used for the selection [13]. Drawing pixel pairs from a Gaussian distribution around the center presents the most promising results. The BRIEF descriptor is a good compromise between speed, storage, efficiency, and recognition rate. Since the process of generating the feature descriptor is based on the information from only a few pixels, the descriptor is noise sensitive. Thus, BRIEF utilizes pre-smoothed patches to reduce the noise sensitivity.

2.2.5 Oriented FAST and Rotated BRIEF (ORB)

ORB presents a combination of an oriented FAST detector with a very fast binary descriptor based on BRIEF [41]. Given that these two techniques have shown promising performances they are both utilized for ORB features. FAST features are affected by the lack of orientation and BRIEF is limited as a consequence of the absence of rotational invariance. ORB introduces a fast and accurate orientation for FAST features and the efficient computation of rotated BRIEF.

The FAST detector utilizes an intensity threshold between a pixel and pixels in a circular boundary around the center. FAST has a large response along the edges. However, it lacks the measurement for cornerness of detected corners. Thus, FAST features are filtered by using the Harris corner measure. First a low threshold is utilized in order to generate a high number of features. Afterwards, the N features with the highest Harris measure are selected. FAST features are scale dependent. ORB employs a scale pyramid of images to obtain scale invariance. The orientation measurement for FAST features is generated by calculating the angle of the offset between the center of the patch and the intensity centroid. This computation is simplified to:

$$\theta = \mathrm{atan2}(\sum_{x,y} yI(x,y), \sum_{x,y} xI(x,y)) \tag{4}$$

This generates orientation aware FAST features. ORB also introduces the rBRIEF, which is a rotation aware BRIEF descriptor. Rotation invariance is achieved by rotating the patches in accordance to the determined orientation. To improve the results of the descriptor the binary tests on pixel pairs are not randomly gen-

erated. Instead, a greedy search algorithm was employed on the PASCAL 2006 data set [18] to determine a good set of binary tests.

2.2.6 Center Surround Extremas (CenSurE)

The CenSurE feature detector is a scale invariant center surround detector [1]. This detector computes features at all scales using pixels in the original image to maintain accuracy. A Harris-Laplace and Hessian-Laplace detector compute the Hessian or Harris measure at all locations and scales. For every detected candidate the Laplacian is computed at all scales. This strategy is adapted by using extrema of a center surround response as an approximation. The CenSurE detector computes a simplified center surround filter at all locations and scales. Afterwards, the local extrema in a neighbourhood are found. Finally, these extrema are filtered by computing the Harris measure and eliminating those with weak corner response.

The CenSurE uses bi-level center surround filters, which is a simple approximation of the Laplacian with the Difference of Gaussians. It multiples the image value by either 1 or -1. A filter that is both faithful to the Laplacian and is fast to compute with integral images needs to be designed. A circular filter is the most faithful to the Laplacian, however, hard to compute. Agrawal et al. proposed the use of an octagon filter for good performance and a box filter for good computation time [1]. We use the STAR detector which is an OpenCV modified version of the CenSurE detector in our experiments [11]. [1] proposed a non maximal suppression in a $3 \times 3 \times 3$ neighbourhood in the image pyramid. The magnitude of the responses is taken as an indication of the strength of the feature for being stable. The greater the strength, the more likely it is to be repeated. Weak responses are likely to be unstable. Furthermore, weak features are discarded using a threshold t_r for the response. To filter out the features that lie along an edge or line, the second moment matrix of the response function at the particular scale is used:

$$\mathbf{H} = \begin{pmatrix} \sum L_x^2 & \sum L_x L_y \\ \sum L_x L_y & \sum L_y^2 \end{pmatrix} \tag{5}$$

L_x and L_y are the derivatives of the response function L along x and y within a window that is linearly dependent on the scale of the particular feature point. This is the scale adapted Harris measure and is different from the Hessian matrix used by SIFT [32] to filter out line responses. Once the Harris measure is computed, its trace and determinant is used to compute the ratio of principal curvatures.

3 Panorama generation

We compare various feature detectors and description methods on panoramic scans, i.e., full 360° scans for 3D scan registration. Since we utilize 2D feature detectors and description methods, panoramic image generation is a necessity. To this end, data sets with reflectance information obtained with a terrestrial 3D laser scanner have been used for the generation of panoramic intensity images. In this paper we emphasize on projecting the spherical scanner data to a 2D panoramic image and the related effects on scan registration. Therefore we have implemented seven different projections.

The scanned environment can be considered as projected to a sphere with the scanner positioned in the center of the sphere. The scanner used in this study is capable of scanning its surrounding with a vertical field of view from $-40°$ to $60°$ with the horizon as latitude $= 0°$. Therefore we have data from a vertical field of view of $100°$ and a horizontal field of view of $360°$. Figure 2 illustrates the portion of the artificial sphere as the scanned environment.

Panorama generation is the process of mapping the points in the spherical coordinates θ, φ, r to the image coordinates x and y. Next, we describe the panoramic projections used in this study.

Figure 2: A scanned environment as an artificial sphere with respect to the field of view of the scanner with the scanner positioned at the center of the sphere.

3.1 Equirectangular

This is the simplest projection which is used in many applications to map a portion of a surface of a sphere to a flat image. With this projection the longitude and latitude are mapped to horizontal and vertical coordinates of a grid with no transformation or scaling applied. In an equirectangular projected image all vertical straight lines remain vertical and straight, however horizontal straight lines will become curves except for the horizon. This projection supports $360°$ in the horizontal field of view and $180°$ in the vertical field of view. Coordinates relate linearly to θ and φ. Poles are stretched to the entire width of the image at the top and bottom edges. The equirectangular projection is a cylindrical equidistance projection which is also called rectangular projection. The transformation equations of this projection is:

$$x = \theta \tag{6a}$$

$$y = \varphi \tag{6b}$$

where the longitude θ and the latitude φ are the spherical coordinates.
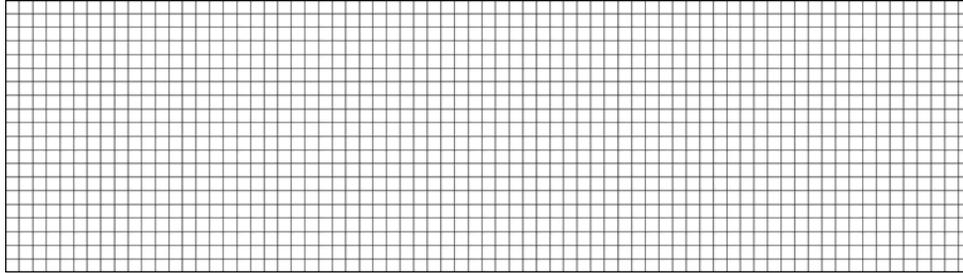
(a) Latitude and longitude map with equirectangular projection

(b) Reflectance map with equirectangular projection

Figure 3: Equirectangular projection

3.2 Cylindrical

This projection is similar to the equirectangular projection and can be envisioned by wrapping a flat piece of paper around the circumference of a sphere, such that it is tangent to the sphere at its equator to form a cylinder around it. Emitting light from the center of the sphere will project the sphere onto the cylinder as shown in Figure 4. In this projection straight vertical lines remain straight, and horizontal lines become curves. It stretches objects vertically, especially the closer they are to the north and south poles of the sphere. In fact this is the reason why the vertical field of view for this projection is restricted to 120° or less. However it has the same recommended horizontal field of view as the equirectangular projection which is a field of view between 120° to 360°.

The vertical and horizontal field of view of our scanner is equal to the recommended range of this projection. The projection process is as simple as for the equirectangular projection. With a vertical field of view from −40° to 60° the no-
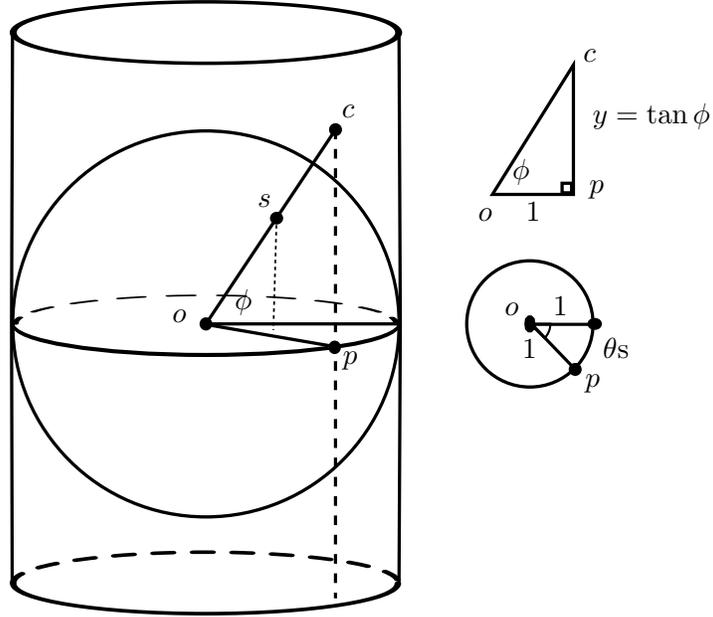
Figure 4: Cylindrical projection diagram.

ticeable distortions of the projection is mostly on top of the mapped image. For outdoor scenarios this is mostly sky. Therefore the distortions have little effect on the feature detection process. To map the panorama data onto the image, the projection proceeds as:

$$x = \theta \tag{7a}$$

$$y = \tan \varphi \tag{7b}$$

which is mapping the longitude θ to the horizontal coordinates and $\tan \varphi$ to the vertical coordinates where φ denotes the latitude. Therefore by utilizing the aforementioned equations and the horizontal and vertical range of the 2D image, the complete scanned 3D data will be mapped onto the image.

3.3 Mercator

The Mercator projection is a conformal projection which is related to the equirect-angular projection and the cylindrical projection. It shows less pronounced distor-

16
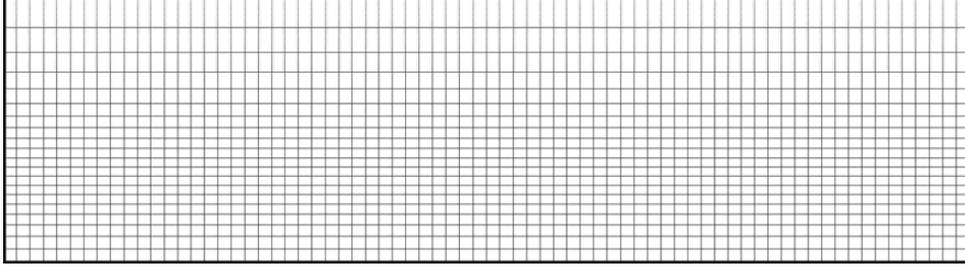
(a) Latitude and longitude map with cylindrical projection

(b) Reflectance map with cylindrical projection

Figure 5: Cylindrical projection

tions compared to the aforementioned projections. It has less vertical stretching and a greater vertical field of view. There is a variation of this projection called transverse Mercator which is used for very tall vertical panoramas. However for the purpose of this paper and due to the fact that our scanner has 100° vertical field of view, we focus on the simple Mercator projection and none of its variations. The Mercator projection is recommended for a horizontal field of view between 120° and 360° and a vertical field of view less than 150°. It is isogonic, i.e., angles are preserved. The projection process proceeds as:

$$x = \theta \tag{8a}$$

$$y = \ln\left(\tan\varphi + \frac{1}{\cos\varphi}\right), \tag{8b}$$

mapping the longitude θ to horizontal coordinates and calculating the vertical coordinates of the image.
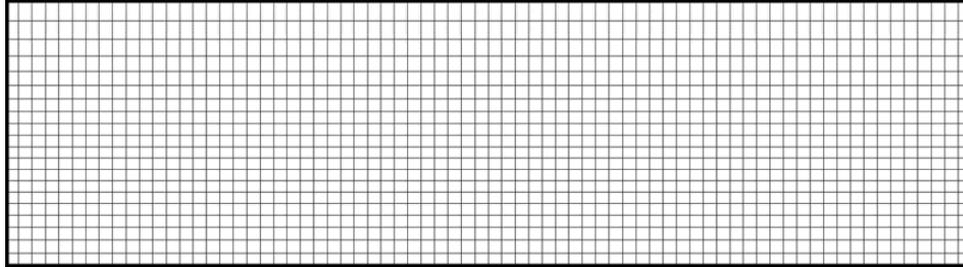
(a) Latitude and longitude map with Mercator projection

(b) Reflectance map with Mercator projection

Figure 6: Mercator projection

3.4 Rectilinear

Rectilinear is a type of projection for mapping only a portion of the surface of a sphere onto a flat image. It is also called "gnomonic" or "tangent-plane" projection. This projection can be imagined by placing a flat piece of paper tangent to a sphere at a single point and illuminating the surface from the center of the sphere. The image projected onto the paper is the rectilinear projection of the portion of the sphere onto the flat image as shown in Figure 7.

The primary advantage of the rectilinear projection is that it maps straight lines in 3D space to straight lines in the 2D image. Its disadvantage is the smaller field of view as compared to the aforementioned projections. Another disadvantage of this projection is that the image is stretched towards the corners

18
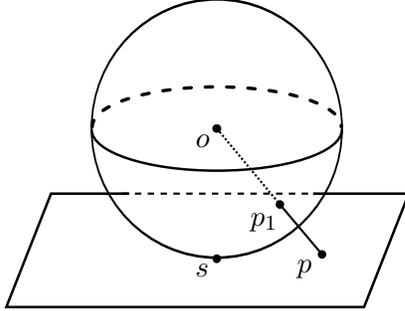
Figure 7: Rectilinear projection diagram.

and the distortion grow with larger fields of view. The projection proceeds as:

$$x = \frac{\cos\varphi \sin(\theta - \theta_0)}{\sin\varphi_1 \sin\varphi + \cos\varphi_1 \cos\varphi \cos(\theta - \theta_0)} \tag{9a}$$

$$y = \frac{\cos\varphi_1 \sin\varphi - \sin\varphi_1 \cos\varphi \cos(\theta - \theta_0)}{\sin\varphi_1 \sin\varphi + \cos\varphi_1 \cos\varphi \cos(\theta - \theta_0)}. \tag{9b}$$

It is recommended to use this projection for a horizontal and vertical field of view less than 120°. Since the vertical field of view of the scanner is less than the recommended vertical field of view of the rectilinear projection, there is no need for a special treatment in the vertical direction. However, the horizontal field of view requires extra processing. We divide the 360° horizontal field of view of the scan into 3 separate parts and pursue the projection with each subset with 120° horizontal field of view. Accordingly the projection of each subset is calculated and mapped to one third of the image.

3.5 Pannini

The Pannini projection, also known as "Recti-Perspective", "Panini" or "Vedutismo", is a mathematical rule for constructing perspective images with a very wide field of view. This projection can be imagined as the rectilinear projection of a 3D cylindrical image. This image is itself a projection of the sphere onto a tangent cylinder. The center of the rectilinear projection can be different and is on the view axis at a distance of d from the cylinder axis (cf. Figure 9). This parameter d can be any non-negative number and determines the projection. When
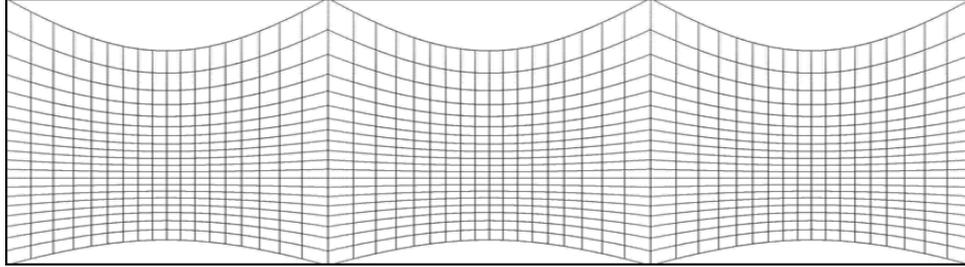
(a) Latitude and longitude map with rectilinear projection

(b) Reflectance map with rectilinear projection

Figure 8: Rectilinear projection

$d = 0$ the projection is rectilinear and when $d \to \infty$ it gives the cylindrical ortho-graphic projection. Therefore, it generates different projections with different d parameters. The projection with $d = 1$ has been defined as the Pannini projection. This projection produces images with a single central vanishing point. The images appear to have correct perspective even when using a wide field of view. The Pannini projection often resembles a normal rectilinear perspective. The horizontal field of view can be very wide, without the distortions near the edges which are noticeable when the rectilinear projection is pushed too far. This is the advantage of the Pannini projection over the rectilinear projection. It keeps straight vertical lines straight and vertical and keeps the radial lines through the center of the image straight. However, it transforms horizontal lines into curves. The recommended field of view for the Pannini projection is less than 150° in both vertical and horizontal directions. Therefore, the projection demands dividing a
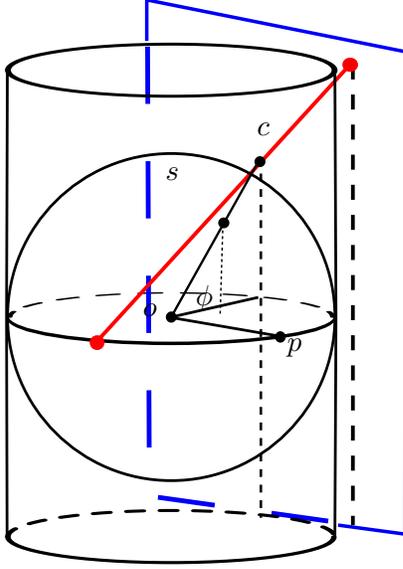
Figure 9: Pannini projection diagram.

$360°$ scan into several sub-sets. The projection proceeds as:

$$x = \frac{(d+1)\sin(\theta - \theta_0)}{d + \sin\varphi_1 \tan\varphi + \cos\varphi_1 \cos(\theta - \theta_0)} \tag{10a}$$

$$y = \frac{(d+1)\tan\varphi\left(\cos\varphi_1 - \sin\varphi_1\left(\frac{1}{\tan\varphi}\right)\cos(\theta - \theta_0)\right)}{d + \sin\varphi_1 \tan\varphi + \cos\varphi_1 \cos(\theta - \theta_0)}. \tag{10b}$$

These are enhanced equations of the Pannini projection that were modified in order to have projection center other than the south pole. This was an assumption for the original Pannini projection equations [44]. Thus θ_0 and φ_1 are the projection center in each subset of the data.

3.6 Stereographic

This projection is a further alternative method for mapping a portion of the surface of a sphere onto a flat image. It can be imagined by placing a flat paper tangent to a sphere and by illuminating it from the opposite pole. Therefore each point on the sphere casts a shadow on the paper. One pole is the center
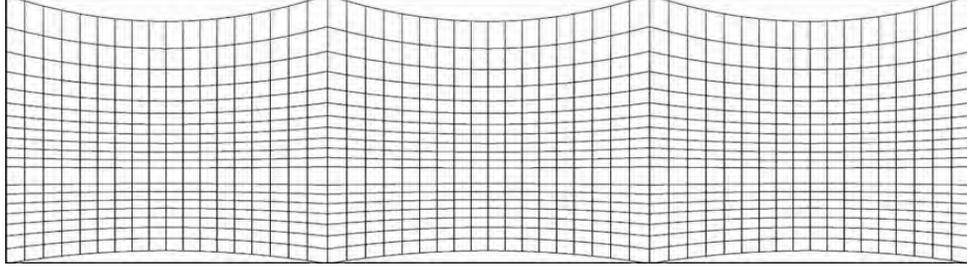
(a) Latitude and longitude map with Pannini projection

(b) Reflectance map with Pannini projection

Figure 10: Pannini projection

of the projection, lines of the latitude appear as circles around the central point. Mapped points gain more distortions further from the pole up to the equator which is increased twice in size compared to the sphere. The north hemisphere is stretched even more through the north pole where it is mapped to infinity. However, the center of projection and the illuminating point are not bound to the poles of the sphere. The center of the projection can be any point on the sphere. The illuminating point can have any distance R from the central point. Images over $330°$ are not very functional. We take this further and utilize a $120°$ horizontal field of view and a $100°$ vertical field of view. In order to satisfy the essentials of this projection we divide the 3D data set into three sub sets, and the projection proceeds as:

$$x = \frac{2R \cos \varphi \sin (\theta - \theta_0)}{1 + \sin \varphi_1 \sin \varphi + \cos \varphi_1 \cos \varphi \cos (\theta - \theta_0)} \tag{11a}$$

$$y = \frac{2R \left(\cos \varphi_1 \sin \varphi - \sin \varphi_1 \cos \varphi \cos (\theta - \theta_0)\right)}{1 + \sin \varphi_1 \sin \varphi + \cos \varphi_1 \cos \varphi \cos (\theta - \theta_0)}, \tag{11b}$$
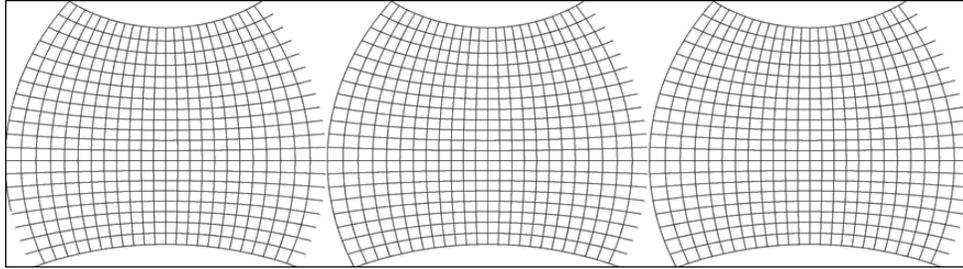
(a) Latitude and longitude map with stereographic projection

(b) Reflectance map with stereographic projection

Figure 11: Stereographic projection

where θ_0 and φ_1 are the projection center in each subset of data. The R parameter is any non negative value. $R = 1$ generates exactly the same equations as the Pannini projection and high values for R introduce more distortion. We used the value 2 for the parameter R in our experiments.

3.7 Z-axis

This projection is not a regular panoramic projection. However, since we have the actual height of the objects in the real world through the sensing process it is reasonable to use those values for the projection purpose as well. We utilize an approximation of the minimum and maximum values of the points in z-axis direction, i.e., in height direction, from the scanned data and the longitude for
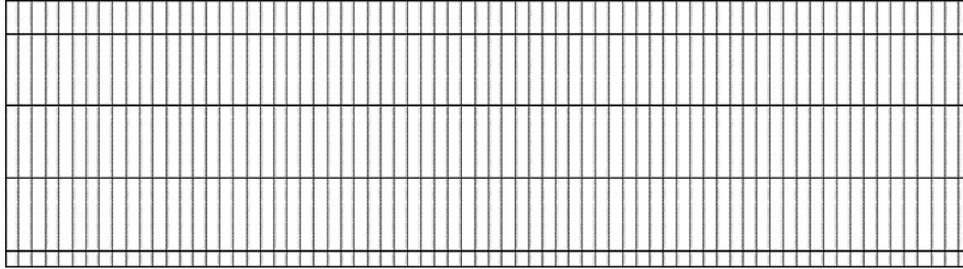
(a) Latitude and longitude map with z-axis projection

(b) Reflectance map with z-axis projection

Figure 12: z-axis projection

the mapping process which proceeds as:

$$x = \theta \tag{12a}$$

$$y = c\,z. \tag{12b}$$

This projection maps the longitude θ to the horizontal coordinates and the height of the points z to the vertical coordinates. c can be any scene dependent constant. Consequently by utilizing the aforementioned equations the 3D data will be mapped to the 2D image.

4   Automatic registration

Registration is the last stage of this process where a RANSAC-like [19] approach is utilized in order to generate the transformation matrix in a pairwise registration scenario. Registration proceeds by taking as input a set of matched features from

two panorama maps. A panorama map stores for each pixel a point with x, y, z and its reflectance. If more than one 3D point falls into one pixel, we save the reflectance and (x, y, z)-coordinate of the 3D point with the largest range. This way we favor points on buildings instead of vegetation [17]. The matching process depends on the type of utilized feature descriptors which must be the same for both scans. Matching means identifying a feature that is closest to a sampled feature based on descriptor comparison. Algorithms such as k-nearest neighbour (KNN), Radius-KNN and brute-force are possible solutions to this problem. The ratio nearest neighbour search as presented by [32] which is a modified version of nearest neighbour search has shown the most promising results. For ratio nearest neighbour search the distances of the feature descriptors to the sampled feature are used. The ratio of the distance of the first closest descriptor to the distance of the second closest descriptor is calculated. A threshold on the ratio determines whether a sample feature has a valid match. These distances are defined as the Euclidean distance between descriptor vectors for the SIFT and the SURF and the Hamming distance in case of the ORB and the BRIEF feature descriptors. Afterwards a RANSAC-like approach is used to filter the outliers of the matching process.

Registration proceeds by testing a subset of combinations of 3 point pair matches and examining the two triangles that are generated by these point pairs. The algorithm calculates the centroid of each triangle and translates the triangles so that their centroids are at the center of the reference frame. The orientation that minimizes the error between the points is computed by the closed form solution proposed by [25]. The rotation is presented in the form of a quaternion which is the eigenvector corresponding to the maximum eigenvalue of the following matrix:

$$
\mathbf{N} = \begin{pmatrix}
S_{xx} + S_{yy} + S_{zz} & S_{yz} - S_{zy} & S_{zx} - S_{xz} & S_{xy} - S_{yx} \\
S_{yz} - S_{zy} & S_{xx} - S_{yy} - S_{zz} & S_{xy} + S_{yx} & S_{zx} + S_{xz} \\
S_{zx} - S_{xz} & S_{xy} + S_{yx} & -S_{xx} + S_{yy} - S_{zz} & S_{yz} + S_{zy} \\
S_{xy} - S_{yx} & S_{zx} + S_{xz} & S_{yz} + S_{zy} & -S_{xx} - S_{yy} + S_{zz}
\end{pmatrix},
$$

(13)

with

$$S_{mn} = \sum_{i=1}^{3} p_{im}^{a} \cdot p_{in}^{b},$$ (14)

where a and b correspond to the first and second triangles respectively, m and n to one of three coordinates x, y or z. Afterwards the rotation matrix R is given by:

$$\mathbf{R} = \begin{pmatrix} a^2 + b^2 - c^2 - d^2 & 2bc - 2ad & 2bd + 2ac \\ 2bc + 2ad & a^2 - b^2 + c^2 - d^2 & 2cd - 2ab \\ 2bd - 2ac & 2cd + 2ab & a^2 - b^2 - c^2 + d^2 \end{pmatrix},$$ (15)

where a, b, c and d are the parameters of the eigenvector $v = (a, b, c, d)^T$ which corresponds to the maximum eigenvalue of N. Consequently the transformation matrix for homogeneous coordinates is given by:

$$\mathbf{M} = \begin{pmatrix} & \mathbf{R} & & \mathbf{t} \\ & & & \\ 0 & 0 & 0 & 1 \end{pmatrix},$$ (16)

where $\mathbf{t}$ is the translation vector and is computed by:

$$\mathbf{t} = \mathbf{R}(-\mathbf{c}_b) + \mathbf{c}_a.$$ (17)

Here, c_a and c_b are the centroids of the triangles a and b. The quality of the transformation matrix M obtained for a pair of triangles is estimated by computing the distances between pairs of matching points. The pairs of points with an error lower than a threshold λ_d are considered as inliers. In all our experiments we set $\lambda_d = 0.5$ m. For transformations with more then 10 inliers we compute the following quality metric, that depends on both the number of inliers n as well as the sum of distances between matching points E :

$$Q = E - In^2.$$ (18)

Here, I is a parameter that scales the dimensionless number of inliers. The transformation matrix with the highest quality is selected as the output of our algorithm.

26

5   Experiments and results

All experiments and tests were carried out on a machine with a Quad-Core processor AMD Athlon^{TM}II X4 640 CPU and 8GB RAM. For the implementation of feature detectors and descriptors and feature matching we utilized the OpenCV library. The remainder of the process was implemented as part of 3DTK – The 3D Toolkit which provides methods to process 3D point clouds and algorithms for high-accurate 6D SLAM [36, 35]. All scans were acquired with a terrestrial laser scanner, the Riegl VZ-400. The scanner was mounted on the mobile robot Irma3D for the acquisition of the outdoor scans in a stop-scan-go fashion. We acquired 3 data sets, 1 small indoor data set and 2 large outdoor data sets. The panorama projected reflectance images of each scan are generated with the proposed panorama projections in section 3. Due to the restricted horizontal field of view of the Pannini, rectilinear and stereographic projections, scans had to be divided into three subsets for these projections. Afterwards, features are detected in each image and the descriptors of features of scan pairs are compared for pairwise matching. We utilized the ratio based matching with a RANSAC-like filtering approach for our experiments with a threshold of 0.8 as a measure for eliminating false matches. Scan registration proceeds on sequential pairwise scans. The success of registration is determined via visual comparison of results with the ground truth. Ground truth was acquired by marker-based manual pre-registration and fine-registration using slam6D from the 3DTK.

We used SIFT, SURF, ORB, STAR, and FAST as feature detectors and SIFT, SURF, and ORB as descriptors which results in 15 possible combinations of these detectors and descriptors. Combining these 15 possibilities with the 7 projections mentioned in section 3 and with two resolutions produces 210 configurations for each scan pair of our data sets.

5.1   Interfaith house (indoor environment)

We acquired two scans from the inside of the Interfaith house located on the campus of the Jacobs University Bremen, Germany. We tested all of the panoramic projections to generate panoramic reflectance images of the 3D point clouds re-

Figure 13: Reflectance image of matched SURF features with the Mercator projection of the Interfaith House of the Jacobs University Campus.

ceived from the scanner. Two different resolutions were used for the panorama image projections. The lower resolution is 3600×1000 and the higher resolution is 5040×1400. Due to the fact that the environment is symmetrical the registration process is difficult. Correct registrations are only obtained with the Mercator projection using the SURF feature detector and descriptor for both resolutions and with the SIFT feature detector and descriptor with low resolution. A correct registration was also generated with the Pannini projection with the SURF feature detector and descriptor for both resolutions. This shows that both the type of projection and the type of feature have significant effect on the registration process.

5.2   Bremen city (outdoor environment)

This data set contains 13 scans from the city center of Bremen, Germany. Scan resolution was set to 0.04 degrees which defines the amount of rotation between each measurement in both vertical and horizontal direction. The scans were acquired with several meters distance between each pair. Figure 14 presents a
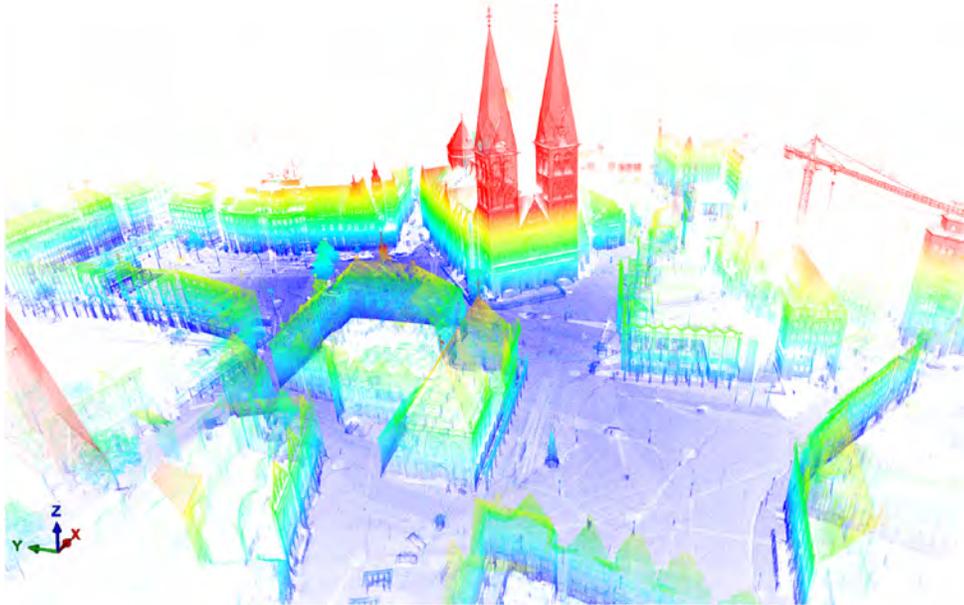
Figure 14: City center of Bremen.

view of the scans after registration.

Only 6 out of all possible configurations generate correct registrations at all. Table 1 displays the percentage of successfully registered subsequent pairs of scans for those configurations. It shows the results with all possible projections with the image resolution 5040×1400 of the Bremen City data set. The FAST feature detector generates a high number of features on each panoramic image. Therefore the matching process, especially with SIFT and SURF descriptors, is extremely slow and ineffective. However, not all of the demonstrated combinations in table 1 are practical since some have poor performance. The combinations with SIFT as detector and descriptor and SURF as detector and descriptor are the most promising.

As seen in table 1, 92% of success could be achieved with SIFT as detector and descriptor and Pannini as projection and additionally with SURF as detector

Table 1: Comparison between the percentage of successful registrations of six detector and descriptor combinations with presented projections. The image resolution was 5040 × 1400. Please note, that 92% success indicates that the entire Bremen City data set is successfully registered except for one pair of scans, namely the loop closing pair.

Detector+Descriptor	Projections						
	EQ	CY	ME	RE	PA	ST	ZA
SIFT+SIFT	84%	69%	76%	30%	92%	76%	15%
SIFT+SURF	38%	38%	61%	30%	46%	46%	7%
SURF+SURF	76%	76%	92%	38%	76%	76%	23%
ORB+ORB	15%	15%	7%	0%	7%	0%	0%
STAR+ORB	15%	7%	7%	0%	7%	15%	0%
STAR+SURF	0%	7%	7%	0%	0%	0%	0%

and descriptor and Mercator as projection. This indicates that the entire Bremen City data set is successfully registered except for one pair of scans, namely the loop closing pair.

Successful registration depends on the feature detection process and the selection of descriptor type together with promising projections. In addition, it also relies strongly on the amount of overlap between scan pairs. The features detected from these overlaps define the inliers in the matching process and a certain amount of inliers are required for generating a correct registration. The number of detected features in an overlap and the uniqueness of detected features are also crucial to the registration results. Since the environment contains symmetric objects, in some circumstances the generation of correct registration is difficult even with a large overlap. Due to the aforementioned facts, scan pairs are divided into three categories. Scan pairs with a large overlap area and a unique set of features, scan pairs with a reasonable overlap yet not enough unique features and scan pairs with tiny overlap and not enough unique features. The first

Table 2: Comparison of the matched features with different projections and SIFT as the feature detector and descriptor with 5040×1400 resolution.

Information	Scan Pair	Projection						
		EQ	CY	ME	RE	PA	ST	ZA
Matches	1 — 2	1413	1219	1388	1565	1566	1604	1802
	5 — 6	1677	1464	1649	2289	1948	1983	1148
	11 — 12	1600	1358	1624	1973	1973	2021	1838
Filtered Matches	1 — 2	41	47	53	22	39	33	30
	5 — 6	26	21	24	27	27	24	50
	11 — 12	91	117	97	77	101	58	33
Inliers	1 — 2	22	27	6	0	15	11	0
	5 — 6	0	0	0	0	7	0	0
	11 — 12	57	94	70	49	87	48	0

category is the easiest and is matchable with most of the configurations. The second category is difficult to match. The third category is the most challenging which generally generates incorrect results even with promising combinations of feature detector, descriptor and projection.

Table 3: Comparison of the matched features with different projections and SURF as the feature detector and descriptor with 5040×1400 resolution.

Information	Scan Pair	Projection						
		EQ	CY	ME	RE	PA	ST	ZA
Matches	1 — 2	3730	3197	3430	4577	4276	4686	8218
	5 — 6	4426	4490	4332	6764	5160	5097	5026
	11 — 12	3856	3125	3600	5108	4924	5277	8666
Filtered Matches	1 — 2	82	78	94	45	90	69	63
	5 — 6	40	43	36	59	44	43	51
	11 — 12	162	130	139	135	153	127	68
Inliers	1 — 2	24	39	39	6	13	39	0
	5 — 6	0	0	6	0	14	0	0
	11 — 12	119	87	98	89	112	85	0

Tables 2 and 3 show the number of matches, filtered matches and inliers for SIFT and SURF as the detector and descriptor for three scan pairs of the Bremen City data set. These pairs are representative examples for each of the three classes of difficulty for scan pairs. In these tables "matches" are the number of matched features based on the ratio nearest neighbour search and "filtered matches" are the number of matches after the RANSAC-like filtering process. The "inliers" are the number of matches that corresponds to the correct registration. The number of matches that the SURF detector generates is greater by roughly the factor of 3 in comparison to the SIFT detector. The quantity of the filtered matches and inliers are therefore higher for SIFT. Figures 15, 16 and 17 present reflectance images of matched features with SURF as the detector and descriptor and 5040×1400 resolution for the demonstrated scan pairs.

As evidenced by both Figure 16 and the corresponding rows of tables 2 and 3 the overlap area in the scan pair 5 - 6 is small, containing a slightly tilted wall and lots of windows. Therefore, registration of this type of scan pairs due to the
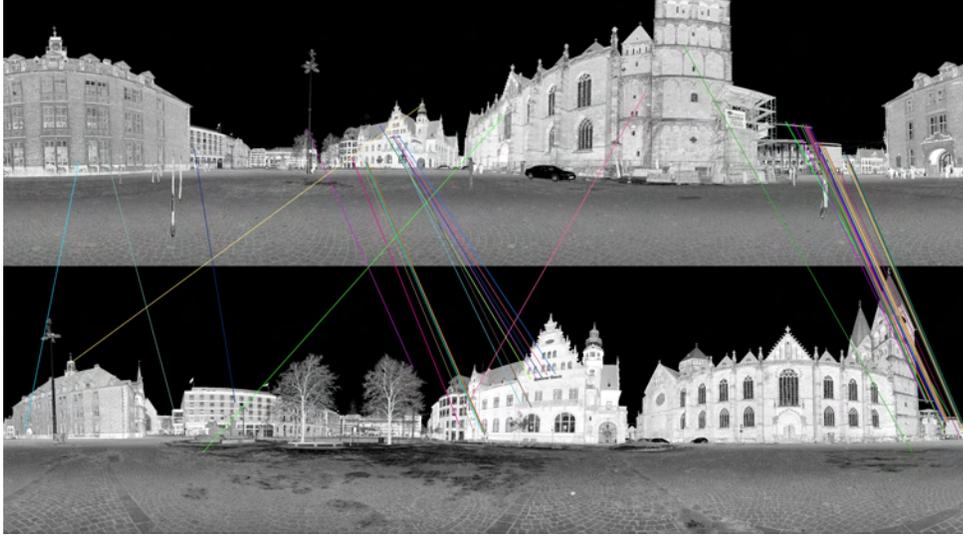
Figure 15: Reflectance image of pair 1 - 2 using the Mercator projection at a resolution of 5040 × 1400. Filtered SURF feature matches are indicated by the lines.

lack of sufficient unique features is a demanding process. In view of the presented facts in tables 2 and 3 the registration of scan pair 5 - 6 is feasible only with the Mercator or Pannini projection.

The most time consuming part of the entire procedure is the registration step. There is a correlation between the amount of filtered matches and the consumed time. Due to the fact that registration time is higher than for the rest of the process by one order of magnitude, the required time depends most strongly on the number of filtered matches. However, the number of filtered matches is in turn dependent on the amount of detected features; both of them rely on the projection and the feature detector methods.

Furthermore, the resolution of the generated panoramic reflectance images from the data set is essential for the feature detection process. It is clear that by increasing the resolution the amount of detected features increases. A larger num-

33
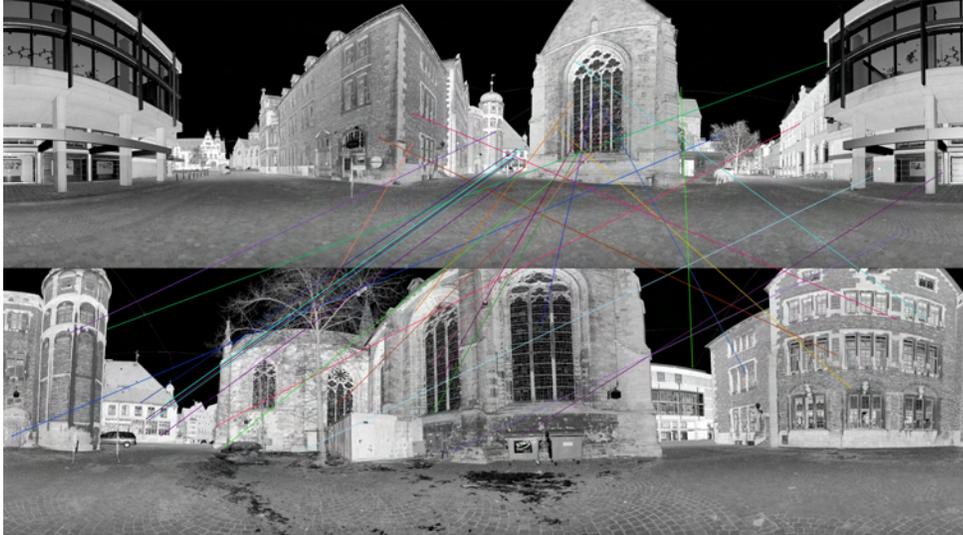
Figure 16: Reflectance image of pair 5 - 6 using the Mercator projection at a resolution of 5040 × 1400. Filtered SURF feature matches are indicated by the lines.

ber of features yields a significantly higher number of filtered matches, and thus results in a more time consuming registration. Similarly lower resolution yields lower number of features and therefore faster registration. Since the amount of filtered matches is crucial to the registration process and a minimum number of filtered matches is essential for successful registration, two resolutions have been used in our experiments to determine a balance between success rate and speed of registration. Table 4 demonstrates the percentage of correct registrations of the Bremen City data set with SIFT and SURF as both detector and descriptor for both resolutions. The success rate by using a larger resolution is higher.

5.3   Campus of Jacobs University (outdoor environment)

This data set contains 122 terrestrial 3D scans from the campus of the Jacobs University Bremen. To acquire this data set the rotation between each measurement was fixed to 0.04 degrees in both horizontal and vertical directions. These scans have been acquired with several meters distance between each scan.
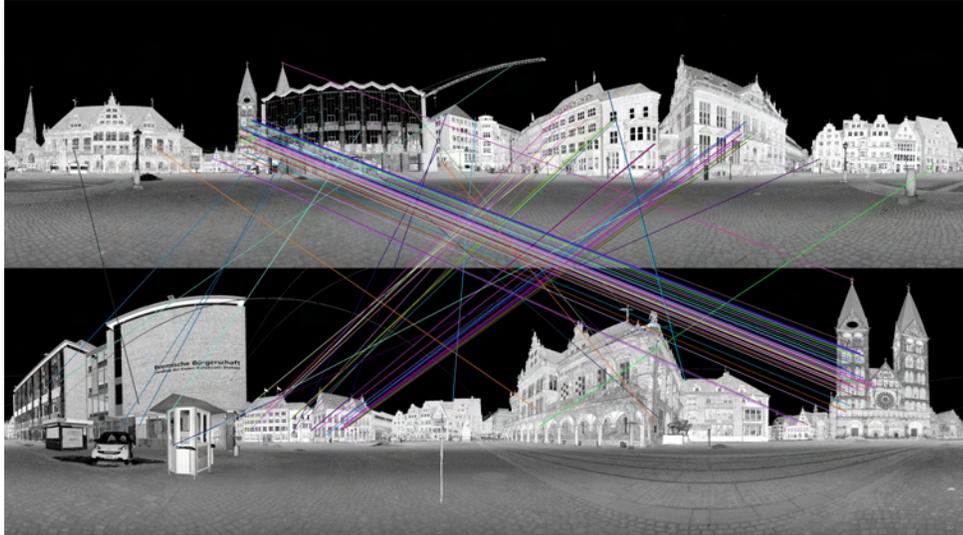
Figure 17: Reflectance image of pair 11 - 12 using the Mercator projection at a resolution of 5040×1400. Filtered SURF feature matches are indicated by the lines.

Figure 18 depicts an orthoimage on a part of the scans after registration, while Figure 19 presents a typical scan in the Pannini projection.

With the results of the Bremen City data set in mind, two combinations of feature detectors and descriptors have been analyzed for the Campus data set: the SIFT feature detector and descriptor and the SURF feature detector and descriptor in combination with all of the presented projections in section 3 and the two proposed resolutions. Table 5 presents the percentage of accurate registrations of the Campus data set for these configurations.

Similar to the results from the Bremen City data set we obtain a higher success rate for the higher resolution. The best projections are Pannini and Mercator which result in a higher success rate with SIFT and SURF respectively. Overall, we obtained a lower success rate on the Campus data set. This is most likely due to the fact that the buildings on the campus are quite symmetrical. Therefore the

35

Table 4: Comparison of the success rate of the most promising combinations of detector and descriptor for all projections and two different resolutions on the Bremen City data set.

Detector+Descriptor	Projection	3600x1000	5040x1400
SIFT+SIFT	Equirectangular	53%	84%
	Cylindrical	69%	69%
	Mercator	76%	76%
	Rectilinear	23%	30%
	Pannini	69%	92%
	Stereographic	76%	76%
	Z-axis	15%	15%
SURF+SURF	Equirectangular	53%	76%
	Cylindrical	30%	76%
	Mercator	46%	92%
	Rectilinear	23%	38%
	Pannini	61%	76%
	Stereographic	61%	76%
	Z-axis	15%	23%

registration process is more challenging. In addition, the environment contains a high amount of vegetation. Trees produce a great quantity of features which results in a significant amount of wrong matches.

6 Conclusions and outlook

This paper presents a survey and evaluation of modern image features and descriptors with respect to the task of 3D scan registration based on the reflectance image of the scans. To this end, we employed panoramic, i.e., 360° scans. We

Figure 18: The campus of the Jacobs University.

investigate the performance of different projections. We present the effects of utilizing different projections with both real-time features and more robust and invariance features. The SURF detector in combination with the SURF descriptor and the SIFT detector in combination with the SIFT descriptor are the best choices as expected for the task of automatic registration, since they are more robust and rotation invariance than real-time features. Furthermore, this paper contributes a study of projection methods for creating a panoramic image of a 3D scan. A realistic, real world evaluation shows that the projection of a 3D scan into a 360° panoramic image for automatic feature based registration is crucial. We showed that the Mercator and the Pannini projection are most suitable: the Mercator projection is an isogonic projection, i.e., angles are preserved; Pannini is also designed to have as few distortions as possible.

Needless to say a lot of work remains to be done. As the experiments show, none of the methods can reliably register all data sets, even though a human
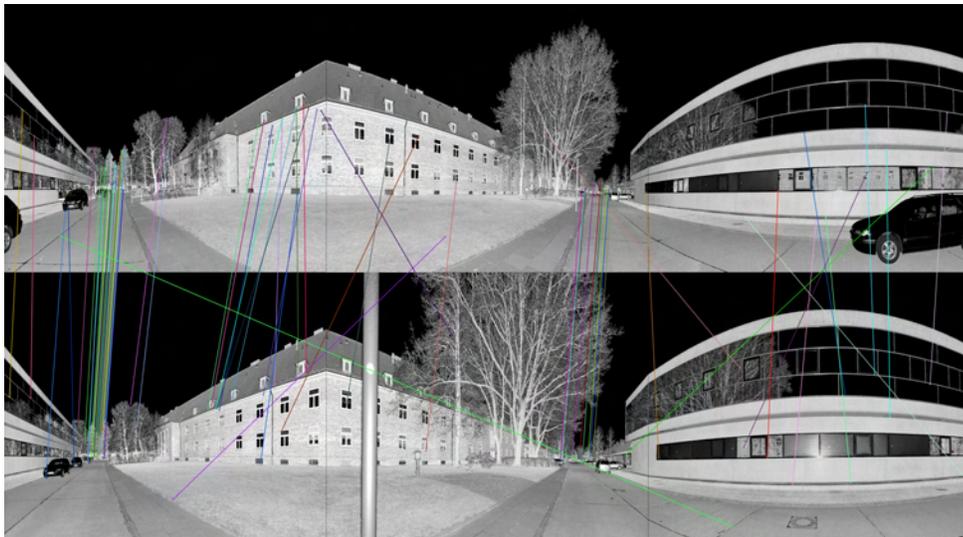
Figure 19: A typical scan from the Campus data set in the Pannini projection.

operator is able to identify the overlapping parts of the scans and can select corresponding features. In future work, we will redesign the matching process by considering a set of similar features and eliminating wrong matches by using structural features, i.e., local 3D surface descriptions like Point Feature Histograms (PFH) [43] or Fast Point Feature Histograms (FPFH) [42].

In future work, we will combine the feature-based registration solution presented here with our feature-less registration methods known as 6D SLAM [36, 35]. This will yield a bundle adjustment solution for 3D laser scans. In addition, we will concentrate on estimating the graph, relating all scan poses.

Acknowledgments

Table 5: Comparison of the success rate of the most promising combinations of detector and descriptor with all projections and two different resolutions on the Campus data set.

Detector+Descriptor	Projection	3600x1000	5040x1400
SIFT+SIFT	Equirectangular	51%	55%
	Cylindrical	46%	53%
	Mercator	51%	55%
	Rectilinear	38%	29%
	Pannini	55%	59%
	Stereographic	55%	59%
	Z-axis	34%	27%
SURF+SURF	Equirectangular	45%	57%
	Cylindrical	38%	39%
	Mercator	44%	61%
	Rectilinear	28%	29%
	Pannini	41%	55%
	Stereographic	37%	52%
	Z-axis	34%	32%

References

[1] Motilal Agrawal, Kurt Konolige, and Morten Rufus Blas. Censure: Center surround extremas for realtime feature detection and matching. In David A. Forsyth, Philip H. S. Torr, and Andrew Zisserman, editors, *ECCV (4)*, volume 5305 of *Lecture Notes in Computer Science*, pages 102–115. Springer, 2008.

[2] D. G. Aguilera, P. R. Gonzalvez, and J.G. Lahoz. An Automatic Procedure for Co-registration of Terrestrial Laser Scanners and Digital Cameras. *ISPRS Journal of Photogrammetry and Remote Sensing*, pages 308–316, 2009.

[3] Shahar Barnea and Sagi Filin. Keypoint based Autonomous Registration of Terrestrial Laser Point-Clouds. *ISPRS Journal of Photogrammetry & Remote Sensing*, 63(1):19–35, 2008.

[4] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, June 2008.

[5] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *9th European Conference on Computer Vision*, pages 404–417, May 2006.

[6] Jeffrey S. Beis and David G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 1000–1006, Washington, DC, USA, 1997. IEEE Computer Society.

[7] Gerhard H. Bendels, Patrick Degener, Roland Wahl, Marcel Körtgen, and Reinhard Klein. Image-based registration of 3d-range data using feature surface elements. In Y. Chrysanthou, K. Cain, N. Silberman, and Franco Niccolucci, editors, *The 5th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST 2004)*, pages 115–124. Eurographics, 2004.

[8] P. Besl and N. McKay. A Method for Registration of 3–D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 14(2):239 – 256, February 1992.

[9] J. Böhm and S. Becker. Automatic marker-free registration of terrestrial laser scans using reflectance features. In *Proceedings of 8th Conference on Optical 3D Measurement Techniques*, pages 338–344, Zurich, Switzerland, July 2007.

[10] O. Booij, B. Terwijn, Z., and B. Krose. Navigation using an Appearance Based Topological Map. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pages 3927–3932, Rome, Italy, May 2007.

[11] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[12] C. Brenner, C. Dold, and N. Ripperda. Coarse orientation of terrestrial laser scans in urban environments. *ISPRS Journal of Photogrammetry & Remote Sensing*, 63(1):4–18, 2008.

[13] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *Proceedings of the 11th European Conference on Computer vision: Part IV*, volume 6314 of *ECCV'10*, pages 778–792, Berlin, Heidelberg, 2010. Springer-Verlag.

[14] Y. Chen and G. Medioni. Object Modelling by Registration of Multiple Range Images. In *Proceedings of the IEEE Conference on Robotics and Automation (ICRA '91)*, pages 2724–2729, Sacramento, CA, USA, April 1991.

[15] M. Cummins and P. Newman. Probabilistic Appearance Based Navigation and Loop Closing. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pages 1828–1833, Rome, Italy, May 2007.

[16] M. Cummins and P. Newman. Accelerated Appearance-only SLAM. In *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '08)*, pages 1828–1833, Pasadena, CA, USA, May 2008.

[17] J. Elseberg, D. Borrmann, and A. Nüchter. Full Wave Analysis in 3D Laser Scans for Vegetation Detection in Urban Environments. In *Proceedings of the XXIII International Symposium on Information, Communication and Automation Technologies (ICAT '11)*, Sarajevo, Bosnia, October 2011.

[18] M. Everingham, A. Zisserman, C. K. I. Williams, and L. Van Gool. The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results, 2006.

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

[20] A. Flint, A. Dick, and A. J. van den Hengel. Thrift: Local 3d structure recognition. In *Proceedings of the 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications (DICTA '07)*, pages 182–188, 2007.

[21] W. Förstner and E. Gülch. A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centers of Circular Features. In *Proceedings of the ISPRS Intercommission Workshop on Fast Processing of Photogrammetric Data*, pages 281–305, 1987.

[22] Wolfgang Förstner, Timo Dickscheid, and Falko Schindler. Detecting interpretable and accurate scale-invariant keypoints. In *12th IEEE International Conference on Computer Vision (ICCV'09)*, pages 2256–2263, Kyoto, Japan, 2009.

[23] Peter Hansen, Peter Corke, Wageeh W. Boles, and Kostas Daniilidis. Scale-invariant features on the sphere. In *ICCV*, 2007.

[24] Peter Hansen, Peter Croke, Wageeh Boles, and Kostas Daniilidis. Scale invariant feature matching with wide angle images. In *In IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2007.

[25] Berthold K. P. Horn. Closed–form Solution of Absolute Orientation using Unit Quaternions. *Journal of the Optical Society of America A*, 4(4):629–642, April 1987.

[26] D. Huber. *Automatic Three-dimensional Modeling from Reality*. PhD thesis, Carnegie Mellon University, 2002.

[27] Zhizhong Kang, Jonathan Li, Liqiang Zhang, Qile Zhao, and Sisi Zlatanova. Automatic Registration of Terrestrial Laser Scanning Point Clouds using Panoramic Reflectance Images. *Sensors*, pages 2621–2646, 2009.

[28] K. Konolige, J. Bowman, J. D. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua. View-based maps. In *Robotics: Science and Systems (RSS '09)*, pages 941–957, Seattle, USA, 2009.

[29] Kevin Köser and Reinhard Koch. Perspectively invariant normal features. In *International Conference on Computer Vision*, pages 14–21, 2007.

[30] Young Jin Lee, Do-Yoon Kim, and Myung Jin Chung. Feature matching in omnidirectional images with a large sensor motion for map generation of a mobile robot. *Pattern Recogn. Lett.*, 25(4):413–427, March 2004.

[31] David G. Lowe. Object Recognition from Local Scale-Invariant Features. In *Proceedings of the International Conference on Computer Vision*, volume 2 of *ICCV '99*, pages 1150–, Washington, DC, USA, 1999. IEEE Computer Society.

[32] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.

[33] Martin Magnusson, Henrik Andreasson, Andreas Nüchter, and Achim J. Lilienthal. Automatic appearance-based loop detection from 3d laser data

using the normal distributions transform. *Journal of Field Robotics (JFR), Special Issue on Three-Dimensional Mapping*, 26(11–12):892–914, November/December 2009.

[34] N. Meierhold, M. Spehrb, A. Schilling, S. Gumhold, and H.-G. Maas. Automatic Feature Matching Between Digital Images and 2D Representations of a 3D Laser Scanner Point Cloud. In *Proceedings of the Commission V Symposium, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVIII, Part 5*, pages 446–451, Newcastle upon Tyne, UK, 2010.

[35] Andreas Nüchter. *3D Robotic Mapping: The Simultaneous Localization and Mapping Problem with Six Degrees of Freedom*. Springer Publishing Company, Incorporated, 2009.

[36] Andreas Nüchter, Kai Lingemann, Joachim Hertzberg, and Hartmut Surmann. 6D SLAM–3D Mapping Outdoor Environments: Research Articles. *J. Field Robot.*, 24(8-9):699–722, August 2007.

[37] Laxmi Parida, Davi Geiger, and Robert Hummel. Junctions: Detection, classification and reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:687–698, 1998.

[38] Kaustubh Pathak, Dorit Borrmann, Jan Elseberg, Narunas Vaskevicius, Andreas Birk, and Andreas Nüchter. Evaluation of the robustness of planar-patches based 3d-registration using marker-based ground-truth in an outdoor urban scenario. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '10)*, pages 5725–5730, Taipei, Taiwan, October 2010.

[39] Edward Rosten and Tom Drummond. Fusing Points and Lines for High Performance Tracking. In *IEEE International Conference on Computer Vision*, volume 2, pages 1508–1511. Springer, October 2005.

[40] Edward Rosten and Tom Drummond. Machine Learning for High-Speed Corner Detection. In *European Conference on Computer Vision*, volume 1, pages 430–443, May 2006.

[41] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *IEEE International Conference on Computer Vision*, pages 2564–2571, Barcelona, November 2011.

[42] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Proceedings of the the IEEE International Conference on Robotics and Automation (ICRA '09)*, pages 1848–1853, Kobe, Japan, May 2009.

[43] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz. Aligning point cloud views using persistent feature histograms. In *Proceedings of the 21st IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*, pages 3384–3391, Nice, France, September 2009.

[44] Thomas K. Sharpless, Bruno Postle, and Daniel M. German. Pannini: A new projection for rendering wide angle perspective images. In *Computational Aesthetics in Graphics, Visualization, and Imaging*, pages 9–16, 2010.

[45] I. Stamos and P. Allen. 3-D Model Construction Using Range and Image Data. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR '00)*, pages 531–536, USA, June 2000.

[46] B. Steder, G. Grisetti, and W. Burgard. Robust place recognition for 3D range data based on point features. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, pages 1400–1405, 2010.

[47] Christoffer Valgren and Achim J. Lilienthal. Sift, surf & seasons: Appearance-based long-term localization in outdoor environments. *Journal Robotics and Autonomous Systems (JRAS)*, 58(2):157–165, February 2010.

[48] X. Wang, C. Toth, D. Grejner-Brzezinska, and H. Sun. Integration of Terrestrial Laser Scanner for Ground Navigation in GPS-challenged Environments.

In *Proceedings of the XXIst ISPRS Congress: Commission V, WG 3*, pages 513–518, 2008.

[49] Zhi Wang and Claus Brenner. Point Based Registration of Terrestrial Laser Data using Intensity and Geometry Features. In *ISPRS Congress ('08)*, pages 583–590, 2008.

[50] Martin Weinmann and Boris Jutzi. Fully automatic image-based registration of unorganized tls data. In *ISPRS Workshop Laser scanning 2011, ISPRS*, 2011.

[51] Martin Weinmann, Michael Weinmann, Stefan Hinz, and Boris Jutzi. Fast and automatic image-based registration of tls data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6, Supplement):62–70, October 2011.

[52] Changchang Wu, Brian Clipp, Xiaowei Li, Jan-Michael Frahm, and Marc Pollefeys. 3d model matching with viewpoint-invariant patches (vip). *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 1–8, 2008.

[53] Z. Zhang. Iterative Point Matching for Registration of Free–form Curves. Technical Report RR-1658, INRIA–Sophia Antipolis, Valbonne Cedex, France, 1992.