

SKYLINE MATCHING BASED CAMERA ORIENTATION FROM IMAGES AND MOBILE MAPPING POINT CLOUDS

Sabine Hofmann*, Daniel Eggert and Claus Brenner

Institute of Cartography and Geoinformatics
Leibniz Universität Hannover
Appelstrasse 9A, 30167 Hannover, Germany
(hofmann, eggert, brenner)@ikg.uni-hannover.de
<http://www.ikg.uni-hannover.de>

Commission ICWG I/Va

KEY WORDS: Laser scanning, Mobile Mapping, Point cloud, Camera, Registration, Matching, Automation

ABSTRACT:

Mobile Mapping is widely used for collecting large amounts of geo-referenced data. An important role plays sensor fusion, in order to evaluate multiple sensors such as laser scanner and cameras jointly. This requires to determine the relative orientation between sensors. Based on data of a *RIEGL VMX-250* mobile mapping system equipped with two laser scanners, four optional cameras, and a highly precise GNSS/IMU system, we propose an approach to improve camera orientations. A manually determined orientation is used as an initial approximation for matching a large number of points in optical images and the corresponding projected scan images. The search space of the point correspondences is reduced to skylines found in both the optical as well as the scan image. The skyline determination is based on alpha shapes, the actual matching is done via an adapted ICP algorithm. The approximate values of the relative orientation are used as starting values for an iterative resection process. Outliers are removed at several stages of the process. Our approach is fully automatic and improves the camera orientation significantly.

1 INTRODUCTION

On Mobile Mapping Systems, different types of sensors are commonly used for the collection of large amounts of data. This data can be used e.g. for visualization, 3D models of large areas, planning tools, mapping and more. Therefore, data integration between different types of sensors needs to be performed in a way that the relative orientation between all sensors is precisely known. As in the example of the *RIEGL VMX-250*, two laser scanners and a GNSS/IMU system are fully integrated and mounted on a single platform, so the relative orientation of these devices is stable over a long period. The IMU system serves as primary coordinate system of the mobile mapping van. In order to achieve high flexibility with the cameras, they are mounted on their own platform where each camera can be oriented as requested. Within each scan project the mounting of the cameras and thus the relative orientation between camera and IMU remains stable. Since the camera mounting is not necessarily stable over a longer period, the relative orientation with reference to the IMU and thus the lidar point cloud, has to be calibrated regularly.

At present, the relative orientation between point cloud and camera is determined manually based on a single image. At least four tie points in a single selectable image and their corresponding 3D points in the lidar point cloud are selected manually and used to determine position and orientation of the camera. One main problem results from the different type of data, meaning that in an optical image corners, e.g. at windows, are good features, whereas there are normally no lidar points directly on corners or edges. Therefore, the manual result might be inaccurate due to the use of only a very few points. Another possible source of inaccuracy is the use of only a single image. In case the chosen 2D points are not well distributed in the selected image, the resulting orientation of the camera might also be inaccurate. Using such an inaccurate calibration matrix leads to a wrong mapping of scan

points to pixel coordinates. As an example see Figure 1, where the right part of the roof is colored using light blue pixels from the optical image which belong to the sky.

Since manual orientation for every scan project is a time consuming task and depends on user abilities, automatic procedures for camera calibration are needed.

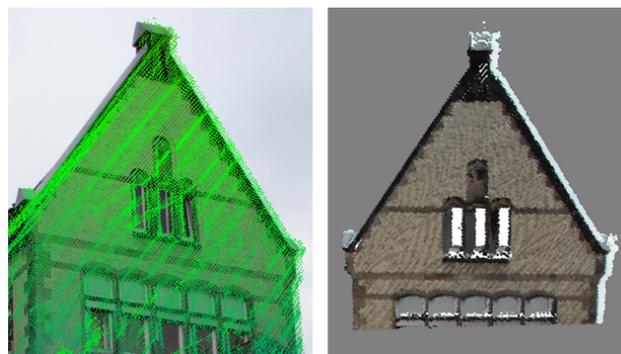


Figure 1: Result of an inaccurate calibration matrix, image overlaid with projected scan points (left) and colored point cloud (right). Points on the right are colored using sky pixels

2 RELATED WORK

In the literature, several approaches for the registration of images and point clouds are presented. (Swart et al., 2011) showed an approach registering a sparse 3D point cloud reconstructed from panoramic images and a dense 3D point cloud from laser scanning. Both 3D point clouds are registered using a dynamic variant of the iterative closest point (ICP) algorithm described by (Besl and McKay, 1992). In addition, images of a virtual camera position are rendered using panoramic images and point clouds. Sub-

*Corresponding author.

sequently, these images are searched for additional corresponding feature points between the datasets. Afterwards, a dynamic ICP is performed again. This approach requires a good geometry between the cameras so that each point is visible in at least two images and ray intersections should not appear under sharp angles. Depending on the field of view and type of the cameras used, this geometric condition might not be met. Additionally, this approach requires 2D image correspondences as well as 3D point correspondences. Furthermore, only images with a sufficient number of matched points can be used.

Feature based approaches are widely used, since no additional information is needed. For example, (Böhm and Becker, 2007) and (Shahzad and Wiggenhagen, 2010) use SIFT features first described by (Lowe, 1999), as corresponding feature points. They use the intensity information of the point cloud in order to compare the feature descriptors. Prerequisite for a successful feature matching between optical and intensity image from a scan is that the point cloud is very dense and intensities from the laser scanner are correlated with intensities of the optical image. Another approach using interest points as features is used by (González-Aguilera et al., 2009) using a hierarchical approach for feature matching based on an area based and least squares matching. Eliminating wrong feature matches is done in an estimation step using the Direct Linear Transform (DLT) and RANSAC, followed by a computation step using least squares adjustment in order to refine the results given from the DLT. (González-Aguilera et al., 2009) also address the problem of the low density of projected points. This is especially important with mobile mapping, since mobile mapping point clouds are often less dense than static terrestrial scans, which causes gaps between scan points when projected to high resolution images. This is also shown by (Wodniok et al., 2013). (Li Chee Ming and Armenakis, 2010) describe an area-based feature approach which is based on corner features, since corners can be naturally found in the data of urban scenes. Corners are defined as intersection of horizontal and vertical edges, while correspondences are detected using cross-correlation. Therefore, parameters such as scale, intensity and orientation need to be similar in the optical and scan image. The *RIEGL VMX-250* also provides intensity values for each point. Though, in our own experiments, we found the intensity to be too noisy to get stable matching results using cross-correlation.

An approach using prior knowledge is matching the outline of objects with a given 3D model. As (Lensch et al., 2001) show, a silhouette based algorithm is usable for small scale objects, where the CAD model of the object is known in advance. In case of mobile mapping, mostly buildings are used as objects. Buildings have several advantages, e.g. 3D models are often available, the outline consists of straight vertical as well as horizontal lines, and the outline normally includes measurements in the upper and lower part of the images. (Haala et al., 2002) and (Haala and Böhm, 2003) for example use the silhouette of buildings extracted from single images using the Generalized Hough Transform and a 3D city model for camera registration. While coarse orientation is done based on low-cost sensors, fine alignment is based on resection using points on the outline of the building and corresponding points from a given CAD model. They found the accuracy of the result depending very much on the geometric and visual quality of the given model and its level-of-detail. In recent years, there is an increasing amount of methods for registration of panoramic or multi-view images which can be found for example in (Pylvanainen et al., 2010) and (Taneja et al., 2012). (Pylvanainen et al., 2010) use panoramic images and a 3D laser scanner together with building outlines from ground plans and an additional height. Their objective is to correct the drift of a mobile mapping system in large areas based on a coarse 3D-model

and non-rigid ICP. Building outlines are segmented using graph cuts in order to obtain shape priors. Therefore, the 3D building outlines are projected into the images. In order to correct mistakes in a single image e.g. caused by occlusions, (Pylvanainen et al., 2010) accumulate their segmentation results over ten images and therefore refine their result. (Taneja et al., 2012) use 3D cadastral models and building outlines for registering spherical panoramic images from StreetView in order to use these images for city planning or tracking of changes in urban areas. The building outline in the first step is segmented using a conditional random field approach where the classifier was trained on manually labeled images. After pose estimation of the camera using the given cadastral building information, the outline extraction is refined using the given prior information. For pose estimation, the projected outline of the 3D model and the given image outline are matched minimizing the number of mismatching outline pixels using Particle Swarm Optimization.

In most of the cited publications (panoramic) images are used. These images normally contain several buildings in different parts of the image and therefore, the points are well distributed. In our case, were images are taken with portrait format vertical to the facade from street level with a DSLR Camera, only small parts of mostly one building, often a part of the roof, are captured. Therefore, matching a given 3D model of the building would not be reliable. Additionally, a 3D model of the scanned area is required. Furthermore, there are many other objects within the scene, such as lamp posts or street signs, which could be used, but normally GIS data do not contain such objects in detail. Using such objects leads to more flexibility than only using building outlines. Furthermore, when using non-panoramic images, an approach is needed to accumulate points from several images until defined geometric constrains for good resection results are met.

3 DATA

Data was collected using a Mobile Mapping System *RIEGL VMX-250* with two laser scanners and four optional cameras as shown in Figure 2.



Figure 2: Mobile mapping van, equipped with the *RIEGL VMX-250*

Position and orientation of the system is gathered using a highly accurate global navigation satellite system (GNSS receiver), an inertial measurement unit (IMU) and an external Distance Measurement Instrument (DMI). All data is preprocessed using corresponding *RIEGL* software and additional software for GNSS/IMU processing.

In the first step, preprocessing obtains a highly accurate trajectory by integrating GNSS base station data and filtering the captured GNSS/IMU data. In the second step a 3D point cloud is

calculated based on trajectory information and the laser scanner measurements using highly accurate time stamps for data integration. Under good conditions an absolute positional accuracy of 2 to 5 cm can be obtained. However, in urban areas the achievable accuracy might be lower due to multipath effects and occlusions from buildings or vegetation which frequently cause GNSS outages. In our experience, an accuracy of 10 to 30 cm in height and up to 20 cm in position has to be expected in urban areas. For more details on the system, see (Riegler LMS GmbH, 2012) and (Rieger et al., 2010).

Each scanner captures full a 360 degree with a maximum rate of 100 profiles and 300 000 measurements per second. This leads to a point density of 2 mm per meter distance along scan profiles and 1 cm per m/s vehicle speed in the direction of travel. Scan data is saved in separate records for each scanner and after each interruption of the scanning process, e.g. after stopping at traffic lights. Lidar measurements are given in a scanner coordinate system with known calibration matrices with respect to the IMU coordinate system. All data, e.g. trajectory, scan points as well as images, are precisely time stamped. Using given time stamps and the known calibration matrices of each scanner, points can be determined in the IMU system. Based on trajectory information all points can be transformed to a global geocentric coordinate system, e.g. ETRS89, or projected to a given virtual camera pose.

For our experiments, a DSLR Nikon D700 camera with 12 MP and a 20 mm Nikkor lens was used for image capturing. During each data recording the mounting of the camera is stable, so the relative orientation between camera and IMU coordinate system is assumed to be identical for all images. The camera is mounted transverse to the direction of travel and therefore, almost perpendicular to facades in portrait format in order to capture the major part of the facade up to the roof of the building in a single image. As an example see Figure 3. Images are taken with a frame rate of two images per second, which leads to an overlap within the image of 20% with an object distance of 5 m and an overlap of approximately 60% with a distance of 10 m at a driving speed of 30 km/h.



Figure 3: Two example images of the Nikon D700 DSLR camera with a Nikkor 20 mm lens

4 APPROACH

Our approach is based on the characteristic that laser scanners do only get range measurements from objects but do not get any response from open sky. Hence, the skyline (that is the contour between foreground objects and the sky) of scanned objects should

correspond to the skyline in an optical image. For this purpose, scan data is projected into a virtual image with size and orientation identical to the captured optical images. Subsequently, all pixels within these scan images, which are covered by a projected point, have known world coordinates. Therefore, corresponding points in optical and scan images can be used for resection. In order to determine the relative orientation between camera and scanner the following steps need to be carried out: image preprocessing in order to get the skyline in both image types, finding corresponding points on these skylines, and finally the iterative robust resection in order to eliminate outliers and calculate the final relative orientation of the camera.

4.1 Skyline Extraction in Optical Images

In order to derive the skyline path from the optical image a threshold is applied to the image. The resulting threshold image is supposed to contain black pixels for objects and white ones for the sky. The needed threshold value is automatically determined based on the image's histogram. Since we intend to separate the bright sky from the darker objects we first find the maximum of the bright values (values above 128) representing the major sky value. As shown in Figure 4 we subsequently find the minimum value between 128 and the previously found maximum.

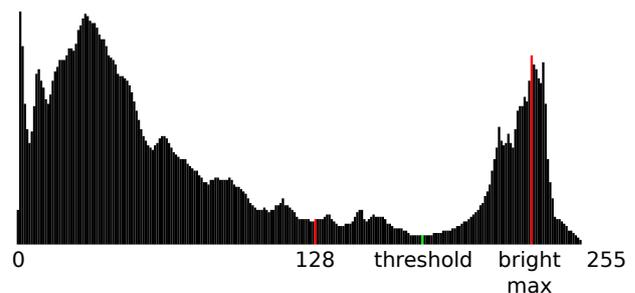


Figure 4: Histogram-based threshold for sky separation

The found value is used as the desired threshold to separate the sky from all world objects. Applying the determined threshold to the optical image results in a binary image as shown in Figure 5 (left). A statistical approach on finding the best threshold in a bimodal histogram of gray scale images is given by (Otsu, 1979), however, we found that our simple scheme works reasonably well on our data. Errors might occur, if a bright facade is illuminated directly by the sun. In this case, parts of the building may also be detected as background. Since that error arises only at some images and the resulting skyline is expectedly rugged, such parts within a skyline will be filtered out in the following steps.

Based on the binary threshold image the alpha shape outline defined by (Edelsbrunner et al., 1983) is derived. Since the point set, in our case all black pixels, is dense and compact, a small disk radius of three pixels is used in order to compute the alpha shape. The resulting outline is visualized in Figure 5 (right). From this alpha shape outline the actual skyline is extracted by removing all parts touching the image border.

The final step of extracting the skyline from the optical image includes the removal of jagged line segments. Those segments mainly originate from tree branches which are no good candidates for scan point matching pixels. Therefore, we propose a window-based approach where for each window it is decided whether it is too jagged and hence shall be removed from the skyline path. There each window ranges over a predefined number of consecutive path points. The bigger that number the bigger the window will be and the more jagged line segments will be

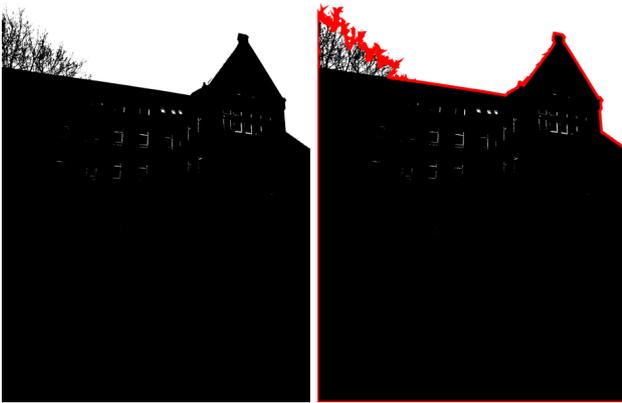


Figure 5: Resulting threshold image (left), alpha shape based outline (right)

removed. In our case a window size of 50 has been found suitable. Within the window the total number of directions between each two consecutive points is counted. The possible directions are binned to eight values: four straight and four diagonal ones. In case six or more directions are present in a window the first half of the windows path points are marked for removal and the window is shifted by half its size along the skyline path. This is repeated until the window has moved all the way to the end of the path.

Instead of removing all marked path points, the connected components to keep or remove are homogenized, where small components are combined with bigger adjacent components. This is necessary in order to remove fairly straight parts of tree branches and keep small building details like chimneys which otherwise would have been kept in the skyline path or removed from it, respectively. The homogenization step repetitively finds the smallest component and merges it with the adjacent ones. In case two components have the same size the component with the bigger adjacent components is chosen for merging. A component is considered small if it is less than four times the window size mentioned earlier. This is repeated until all small components are merged. The result as shown in Figure 6 is a skyline path (green) with most of the jagged path segments removed (red). In case the skyline path of an optical image is reduced to less than 15% of its original size the entire image is excluded from the process since its skyline might contain almost only tree branches.

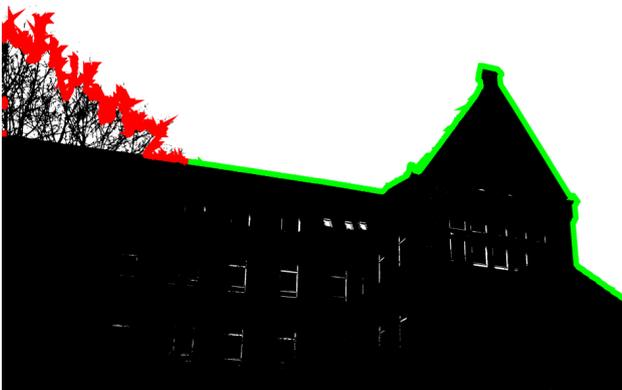


Figure 6: Final skyline with jagged line-segments removed

4.2 Skyline Extraction in Lidar Point Clouds

For the subsequent matching step the skylines path of the corresponding scan point image is needed as well. The scan point images contain all recorded laserscan points that can be projected into each of the optical images taken during the data acquisition. In general a scan point image contains the pixel coordinates of a laserscan measurement and its xyz coordinates, encoded in the images three color channels. As a trade off between accuracy and storage each channel is encoded with four byte allowing single-precision floating point values. Since a scan point image is rather sparse the stored image data is compressed using a lossless compression algorithm like LZ4. For developing purposes all scan point images have been generated in advance but might also be generated on-the-fly for the current optical image.

In principle the scan images are processed in the same way as the optical images. Since there are no lidar measurements for the sky, the initial threshold step can be omitted. For the alpha shape, a bigger disk radius has to be used because the pixels distribution is much sparser compared to the optical image. Furthermore all outline segments without a corresponding optical image skyline counterpart are removed, which makes the jagged line segment removal omissible. The respective results are shown in Figure 7 (left and right).

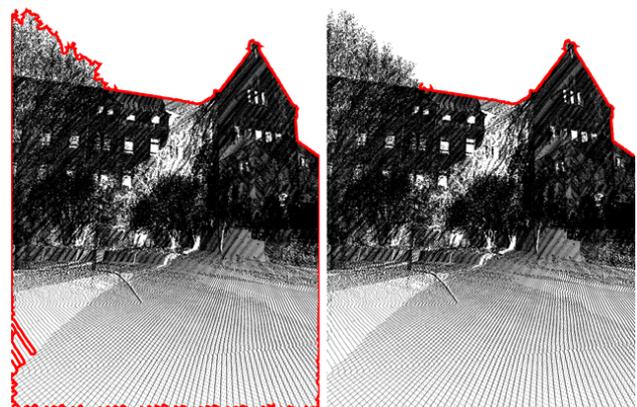


Figure 7: Alpha shape based outline of projected scan point image (left), resulting skyline after removal of unrelated segments (right)

4.3 2D Point Matching

Based on the given skyline, corresponding points in 3D scan image and optical image need to be determined. This is done on separate contiguous clusters of points which belong to the same object or at least belong to one distance class.

In a first step, all skyline pixels of the scan point image are clustered based on their 3D distance to the camera, because the influence of erroneous transformation parameters differs depending on their distance to camera and their position in the image. This clustering allows to obtain better results in the following ICP step. The clustering is based on the Jenks natural breaks classification algorithm invented by (Jenks and Caspall, 1971). The number of distance classes depends on the range of distance within each image. This ensures that points belonging to the same object are in a single point cluster. We have chosen the number of distance classes as

$$num_{classes} = \ln(distanceRange), \quad (1)$$

where $distanceRange$ [m] is determined using maximum and minimum point distance from camera. This leads to a sufficient number of distance classes in case the covered 3D distance is small. In case the distance range is large the number of buckets is growing more slowly. Subsequently, in order to get contiguous clusters, clusters are subdivided if the 2D distance between consecutive points in one cluster is too big. In order to achieve the goal of compact and reliable clusters, small clusters are removed from the skyline. In our case, clusters with less than ten points were eliminated.

In the following step the ICP algorithm is performed in the image plane for each point cluster separately. Point correspondences are only accepted if they are within a selectable 2D distance in order to avoid outlier matches between widely spaced points. Additionally, clusters with too few correspondences on the image skyline, are again eliminated. In our case we used a threshold of less than 50% matched points within a cluster. Furthermore, in the remaining clusters all points without correspondence are also removed. The ICP algorithm terminates if the change in the transformation parameters is smaller than 0.01 pixels.

As illustrated in Figure 8, a simple ICP algorithm would often lead to wrong correspondences, especially at the outline of small objects. As an example a lamp post whose skyline points split into two distance clusters is shown in Figure 8 (left). Since both clusters are close to the same skyline segment from the optical image, the standard ICP algorithm would match both clusters to the same segment as illustrated in Figure 8 (center). Therefore, the ICP algorithm needs to be extended. An overview on different types and extensions of the ICP algorithm is given in (Rusinkiewicz and Levoy, 2001), e.g. using point normals. For each point on the skyline the corresponding point normal pointing away from the object is known and can be used to find the correct correspondences. Hence, only points with similar point normals are matched. In the given example, it forces the left cluster to correctly match with the left side of the lamp post even though those are not the closest points. The corresponding result is shown in Figure 8 (right).

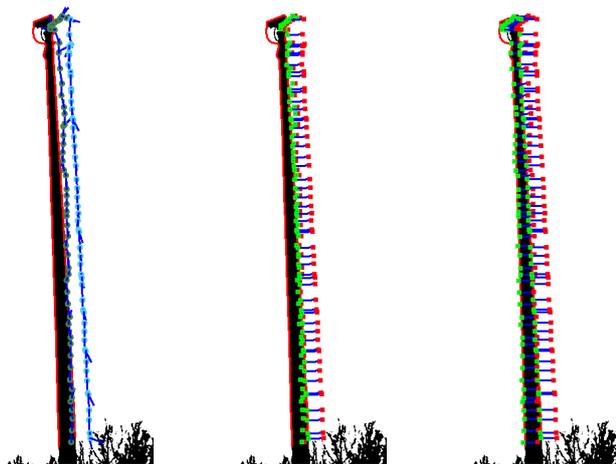


Figure 8: Two distance clusters at a lamp post (left), ICP result without point normal consideration (center), ICP result with point normals (right)

4.4 Image Selection

For a stable result of the resection it is critical to use an appropriate number of points which are sufficiently distributed in 2D image space as well as in 3D object space and, therefore, result in a good geometry. Points should be uniformly distributed

throughout the entire image and cover a large depth of field. This is expressed with the following conditions: a minimum number of images have to be used, points should cover a wide range of 3D distances, and points should cover the four image quadrants evenly.

Since each scan project contains far more images than what is required for a resection, a subset has to be chosen from all recorded images along the trajectory. At the moment, this is done by randomly picking a minimum number of images along the track, but one can think of other sampling methods. In addition to the minimum number of images, the condition that points of a wide range of 3D distances are included must be met. This is realized defining a fixed set of distance buckets and filling them with each newly picked image. In our example we used a maximum distance of 200 m and five distance buckets. Therefore, in each randomly picked optical image the skyline is determined as described in section 4.1. Due to the characteristic of the captured area more than 50% of the images contain mainly trees, which are not useable for our approach and are identified and omitted during skyline extraction. New images are added as long as the condition of points in all distance buckets is not satisfied or the minimum number of images has not been reached. As we use the skyline of the objects, points are more likely to be in the upper part of the image and accordingly we do not use any fixed condition on 2D point distribution. Figure 9 shows the resulting point distribution in 2D image space of one run using ten randomly picked images for calculation.

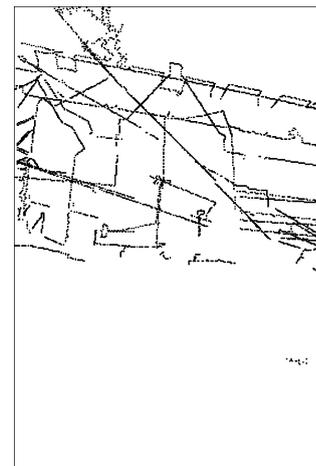


Figure 9: 2D point distribution used for resection

4.5 Resection

In the subsequent processing step, resection is performed in order to refine the relative orientation between camera and the IMU. As functional model the collinearity equations are used:

$$\begin{aligned} x' &= x_0 + c \frac{r_{11}(X - X_0) + r_{12}(Y - Y_0) + r_{13}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)} \Delta x' \\ y' &= y_0 + c \frac{r_{21}(X - X_0) + r_{22}(Y - Y_0) + r_{23}(Z - Z_0)}{r_{31}(X - X_0) + r_{32}(Y - Y_0) + r_{33}(Z - Z_0)} \Delta y' \end{aligned} \quad (2)$$

Image coordinates x', y' on the skyline of the optical image are used as observations. Rotation angles which can be computed from the elements of the rotation matrix r_{ij} and 3D coordinates of the cameras projection center X_0, Y_0, Z_0 are unknowns. Approximate values for all unknowns are given from a manual orientation of the cameras using four manually chosen corresponding points. The interior orientation parameters of the camera

$\Delta x'$, $\Delta y'$ and c are determined by a separate calibration procedure.

Equation 2 can be linearized using Taylor expansion, since approximations of the unknowns are given. This leads to the final linearized equation of the functional model in the form of

$$l + \hat{v} = A \cdot \hat{x}, \quad (3)$$

with observations $l = L - L^0$ (where L are the observations in the optical image and L^0 are calculated from approximate values of the unknowns), corrections v for each observation, design matrix A containing the partial derivatives and finally supplements to the unknown \hat{x} . A more detailed description of the least squares method is given in (Luhmann, 2003). Objective of the adjustment is to minimize the error v :

$$v^T \cdot P \cdot v \rightarrow \min. \quad (4)$$

In the stochastic model all observations are assumed to be equal and independent, therefore the weight matrix P is the identity matrix.

The adjustment is performed iteratively eliminating coarse outliers, generating more precise approximations in each iteration. An outlier test is performed in order to remove remaining outliers from the data set. Therefore, the Data-Snooping-Method described by (Baarda, 1968) indicates observations as outliers based on their normalized error w_i . The normalized error is given by

$$w_i = \frac{v_i}{\hat{s}_{v_i}} \quad (5)$$

with the standard deviation \hat{s}_{v_i} of error v

$$\hat{s}_{v_i} = \hat{s}_{l_i} \cdot \sqrt{(Q_{vv})_{ii}}, \quad (6)$$

and the cofactor matrix Q_{vv} as

$$Q_{vv} = Q_{ll} - A \cdot Q \cdot A^T \quad (7)$$

of the error v . Starting with the maximum normalized error w_i , all observations with $w_i > k$ are eliminated iteratively as outliers with a threshold of $k = 2.56$ as proposed by (Luhmann, 2003).

5 RESULTS AND DISCUSSION

Our algorithm was tested on data of one scan project in the city of Hanover, Germany. The data set contains 977.4 million scan points. Overall, 3385 images per DSLR camera were captured from different types of areas such as highways and urban areas. Along highways, mainly trees form the skyline of the foreground objects as illustrated in Figure 10 (left). In urban areas buildings and road furniture such as traffic lights, lamp posts, or street signs are more prominent and therefore, lead to more linear skylines and a large field of depth within the images which is required for a stable resection. The following results were repeatedly produced with randomly selected images.

Applying our approach on the used data set the initial calibration parameters were shifted by 12.4 cm along track, 1.6 cm across track and 0.2 cm in vertical direction. The changes in angles were largest around the vehicle's x-axis with $\Delta roll = 1.2^\circ$. The angle around the vehicle's z-axis pointing downwards was adjusted with $\Delta yaw = 0.7^\circ$ and the angle around the y-axis with $\Delta pitch = 0.07^\circ$. This leads to a significant improvement of the registration which can be visually determined by projecting scan points to an image using the relative orientation determined by resection. For example, see Figure 11 showing the projected points



Figure 10: Two example images with no suitable skyline

based on the initial calibration matrix and after applying the relative orientation given by resection. The projected scan points are colored using their elevation above ground. As expected the 2D translation between corresponding points in all image regions is smaller than before.

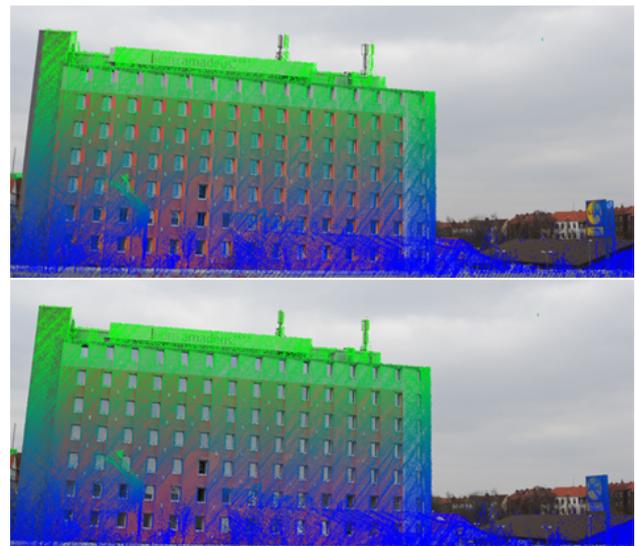


Figure 11: Projected point cloud using the original calibration matrix (top) and after applying the relative orientation based on resection (bottom)

A detail view of Figure 11 is given in Figure 12 showing the projected point cloud based on the original calibration matrix and the resulting image when using the relative orientation given by resection.

Figure 13 illustrates the resulting error when the point cloud is colored using the initial calibration parameters (bottom left). The colors are taken at the projected pixel position as shown in Figure 13 (top left). The sign in Figure 13 has a pixel error of approximately 26 pixels and an error of approximately 1 m in world coordinates. Using the relative orientation based on resection results in a visually considerably better outcome as illustrated in Figure 13 (right). Still, the orientation parameters need refinement as can be seen on the left side of the depicted sign, since there is an remaining error of approximately 3 to 5 pixels in image space. This could be because the interior parameters of the

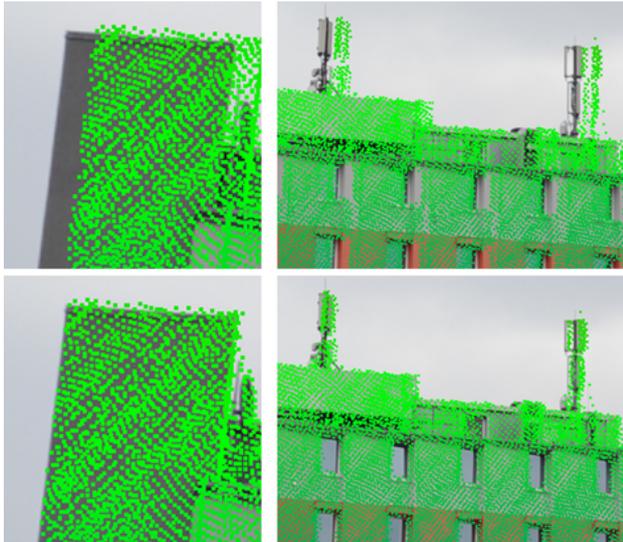


Figure 12: Projected point cloud using the initial relative orientation (top) and the relative orientation given by resection (bottom)

camera are calibrated in advance, but are not stable over a long period and are not calibrated on-the-fly. Similarly, this could occur if there are undetected errors in point correspondences which causes inaccuracies in resection.

Another example, illustrating the results at the roof of a building is given in Figure 14. On the left, wrongly projected and colored points demonstrate the effect of inaccurate calibration parameters whereas the refined relative orientation lead to correctly colored point clouds (right).

Furthermore, the quality of the resection result can be specified using the residuals of the ICP algorithm comparing initial values with the final result executing the ICP algorithm again. Using the original calibration parameters the ICP algorithm results in a 2D transformation of 1.4 pixels in x and -24.3 pixels in y direction with a length of 24.4 pixels on a point cluster with 70 m distance to the camera. After applying our resection approach the ICP algorithm results in a 2D transformation of 0.5 pixels in x and -5.6 pixels in y direction with a length of 5.6 pixels for the same cluster. The ICP results for the given example are visualized in Figure 15.

In the previously discussed examples, the camera pointed to the driver's side of the mobile mapping van and therefore, the distance between camera and facade is reasonably large. Applying our approach to images with the camera directing to the opposite side of the mobile mapping van, indicates that the distance between camera and captured objects has a major impact on the quality of the result, since the distance between camera and facades is usually shorter on the passenger side. In case the distance between camera and objects is too short, often no skyline is visible in the image as depicted in Figure 10 (right). Therefore, image selection is more challenging and hence, more images are used in order to determine the final resection result. Using this mounting, approximately 60% to 70% of the images are rejected during skyline extraction. As mentioned above, in our experiments about 50% of the images were rejected due to trees as shown in Figure 10 (left), the rest is rejected due to invisible sky. Nevertheless, our approach still provides a stable and visually inspected good result for the relative orientation.

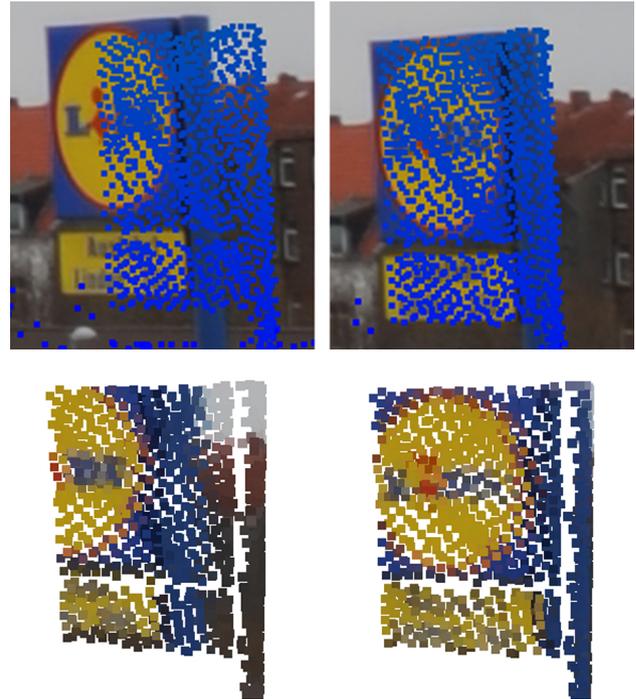


Figure 13: Projected (top) and colored (bottom) point cloud using original calibration (left) and relative orientation given by resection (right)



Figure 14: Projected (top) and colored (bottom) point cloud using original calibration (left) and relative orientation given by resection (right)

6 CONCLUSIONS AND FUTURE WORK

In this work we showed an approach to improve the accuracy of the initially known relative orientation between camera and laser scanners of a mobile mapping system *RIEGL VMX-250*. Matching between optical image and point cloud is done by finding

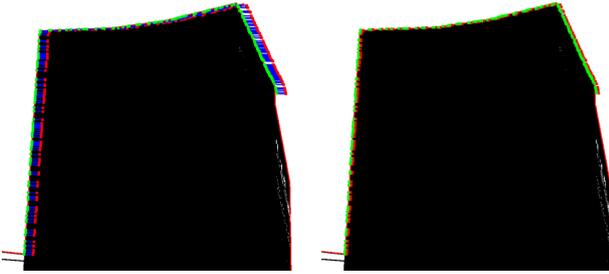


Figure 15: Results of the ICP algorithm using original calibration (left) and relative orientation refined by resection (right)

corresponding points on the skyline of captured objects, which is extracted using alpha shapes. Potential outliers, such as points on trees, are eliminated at this stage. The skyline extraction is followed by a 2D point matching algorithm using an adapted ICP algorithm. The adapted ICP algorithm takes into account the point normals on the skyline, which is similar for corresponding parts of an object. Subsequently, the relative orientation between camera and IMU is improved using resection. As our results show, the alignment of camera and point cloud using resection is visually considerably better than before. Taking the residuals into account the resection leads to an improvement of approximately 20 pixels in the image plane compared to the initial parameters.

Future work needs to be done in order to get a better point distribution in the image plane. Since we use only the skyline of objects, the points occur mainly in the upper part of the images, which results in a suboptimal geometry for resection. Therefore, additional edges should be used in order to evenly distribute the points. Such edges could be the lower parts of lamp posts and buildings, and the curbside along the road. Thus, for the matching process the contour of those objects in the 3D point cloud needs to be determined. Furthermore, the image selection procedure should be varied. Instead of randomly picking images along the route, further experiments are needed to assess the effects of selecting images from different spatial or temporal regions. For example, excluding images captured when the vehicle was turning could remove effects caused by an improper compensation of large roll and pitch angles of the vehicle.

REFERENCES

- Baarda, W., 1968. A testing procedure for use in geodetic networks. Netherlands Geodetic Commission, Delft.
- Besl, P. J. and McKay, N. D., 1992. A method for registration of 3-D shapes. 14(2), pp. 239–256.
- Böhm, J. and Becker, S., 2007. Automatic marker-free registration of terrestrial laser scans using reflectance features. In: Proceedings of 8th Conference on Optical 3D Measurement Techniques, pp. 338–344.
- Edelsbrunner, H., Kirkpatrick, D. and Seidel, R., 1983. On the shape of a set of points in the plane. IEEE Transactions on Information Theory 29(4), pp. 551–559.
- González-Aguilera, D., Rodríguez-Gonzálvez, P. and Gómez-Lahoz, J., 2009. An automatic procedure for co-registration of terrestrial laser scanners and digital cameras. *PandRS* 64(3), pp. 308–316.
- Haala, N. and Böhm, J., 2003. A multi-sensor system for positioning in urban environments. *ISPRS Journal of Photogrammetry & Remote Sensing* 58, pp. 31–42.
- Haala, N., Böhm, J. and Klinec, D., 2002. An integrated approach for the orientation of terrestrial outdoor scenes. *GIS - Zeitschrift für raumbezogene Informationen und Entscheidungen* 12, pp. 40–46.
- Jenks, G. F. and Caspall, F. C., 1971. Error on choroplethic maps. definition, measurement, reduction. *Annals of the Association of American Geographers* 61, pp. 217–244.
- Lensch, H., Heidrich, W. and Seidel, H.-P., 2001. A silhouette-based algorithm for texture registration and stitching. *Graphical Models* 63(4), pp. 245–262.
- Li Chee Ming, J. and Armenakis, C., 2010. Fusion of optical and terrestrial laser scanner data. In: *CGC10*.
- Lowe, D. G., 1999. Object recognition from local scale-invariant features. In: *Proceedings of the International Conference on Computer Vision*, pp. 1150–1157.
- Luhmann, T., 2003. *Nahbereichsphotogrammetrie – Grundlagen, Methoden und Anwendungen*. Herbert Wichmann Verlag, Heidelberg.
- Otsu, N., 1979. A threshold selection method from gray-level histograms. *Systems, Man and Cybernetics, IEEE Transactions on* 9(1), pp. 62–66.
- Pylvanainen, T., Roimela, K., Vedantham, R., Itaranta, J. and Grzeszczuk, R., 2010. Automatic alignment and multi-view segmentation of street view data using 3d shape prior. In: *Fifth International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*, Paris, France.
- Rieger, P., Studnicka, N., Pfennigbauer, M. and Ullrich, A., 2010. Advances in mobile laser scanning data acquisition. In: *FIG congress 2010*, Sydney, Australia.
- Riegl LMS GmbH, 2012. Data Sheet - Riegl VMX-250.
- Rusinkiewicz, S. and Levoy, M., 2001. Efficient variants of the icp algorithm. In: *Third International Conference on 3-D Digital Imaging and Modeling, 2001. Proceedings.*, pp. 145–152.
- Shahzad, S. and Wiggenhagen, M., 2010. Co-registration of terrestrial laser scans and close range digital images using scale invariant features. *Allgemeine Vermessungsnachrichten* 6(117), pp. 208 – 212.
- Swart, A., Broere, J., Veltkamp, R. and Tan, R., 2011. Non-rigid registration of a panoramic image sequence to a lidar point cloud. In: *Proceedings of the 2011 ISPRS conference on Photogrammetric image analysis (PIA'11)*, Springer-Verlag, Berlin, Heidelberg, pp. 73–84.
- Taneja, A., Ballan, L. and Pollefeys, M., 2012. Registration of spherical panoramic images with cadastral 3d models. In: *Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, pp. 479–486.
- Wodniok, J., Hofmann, S., Brenner, C. and Luhmann, T., 2013. Automatische bestimmung der kameraorientierung eines lidar mobile mapping systems. *Allgemeine Vermessungs-Nachrichten* 120(11-12), pp. 375 – 380.