

# ALS-AIDED NAVIGATION OF HELICOPTERS OR UAVs OVER URBAN TERRAIN

M. Hebel<sup>a</sup>, U. Stilla<sup>b</sup>

<sup>a</sup> Fraunhofer Institute of Optronics, System Technologies and Image Exploitation IOSB, 76275 Ettlingen, Germany - marcus.hebel@iosb.fraunhofer.de

<sup>b</sup> Photogrammetry and Remote Sensing, Technische Universität München, 80290 München, Germany - stilla@tum.de

**KEY WORDS:** Airborne laser scanning, LiDAR, GPS/INS, on-line processing, terrain-based navigation, urban data

## ABSTRACT:

Airborne laser scanning (ALS) of urban regions is commonly used as a basis for subsequent city modeling. In this process, data acquisition relies highly on the quality of GPS/INS positioning techniques. Typically, the use of differential GPS and high-precision GPS/INS postprocessing methods are essential to achieve the required accuracy that leads to a consistent database. Contrary to that approach, we aim at using an existing georeferenced city model to correct errors of the assumed sensor position, which is measured under non-differential GPS and/or INS drift conditions. Our approach accounts for guidance of helicopters or UAVs over known urban terrain even at night and during frequent loss of GPS signals. We discuss several possible sources of errors in airborne laser scanner systems and their influence on the measured data. A workflow of real-time capable methods for the segmentation of planar surfaces within ALS data is described. Matching planar objects, identified in both the on-line segmentation results and the existing city model, are used to correct absolute errors of the sensor position.

## 1. INTRODUCTION

### 1.1 Problem description

Airborne laser scanning usually combines a LiDAR device (light detection and ranging) with high-precision navigational sensors (INS and differential GPS) mounted on an aircraft. Range values are derived from measuring the time-of-flight of single laser pulses, and scanning is performed by one or more deflection mirrors in combination with the forward moving sensor platform. The navigational sensors are used to obtain the 3D point associated with each range measurement, resulting in a georeferenced point cloud of the terrain. Laser scanning delivers direct 3D measurements independently from natural lighting conditions, and it offers high accuracy and point density.

A well-established application of laser point clouds acquired at urban areas is the generation of 3D city models. However, the overall precision of the derived city model highly depends on the accuracy of the data input, which is directly dependent on the exactitude of the navigational information. Great efforts are usually required during data acquisition and postprocessing in order to achieve high fitting accuracy of multiple ALS datasets (e.g. neighboring strips). While ALS data acquisition is commonly done to supply other fields of studies with the necessary data, few examples can be found where laser scanners are used directly for pilot assistance. One of these examples is the HELLAS obstacle warning system for helicopters (Schulz et al., 2002), which is designed to detect wires and other obstacles for increased safety during helicopter missions.

Despite increasing performance of LiDAR systems, most remote sensing tasks that require on-line data processing are still accomplished by the use of conventional CCD or infrared cameras. Typical examples are airborne monitoring and observation devices that are used for automatic object recognition, situation analysis or real-time change detection. Utilization of these sensors can support law enforcement, disaster management, and medical or other emergency services.

At the same time, it is often desirable to assist pilots with obstacle avoidance and aircraft guidance in case of poor visibility conditions, during landing operations, or in the event of GPS dropouts. Three-dimensional information as provided by the LiDAR sensor technology can ease these tasks, but the existence of differential GPS ground stations and the feasibility of comprehensive data analysis are not to be considered for these real-time operations.

### 1.2 Overview

The approach of using ALS information to provide on-line navigation support for aircraft guidance over urban terrain is opposite to the process of city model generation. In contrast to the demand for high-precision positioning techniques, it is assumed that a proper georeferenced city model is already available. ALS measurements and matching counterparts in the city model can be taken into consideration if additional navigational information is needed, for example in cases of degraded GPS positioning accuracy.

This paper presents a workflow of methods for the segmentation of planar surfaces in ALS data that can be accomplished in line with the data acquisition process. Since most of currently used airborne laser scanners, like the RIEGL LMS-Q560, measure range values in a pattern of parallel scan lines, the analysis of geometric features is performed directly on this scan line data. Straight line segments are connected across consecutive scan lines to result in planar surfaces. In order to exploit the on-line segmentation results for aircraft navigation, a previously generated set of planar surfaces characterizing the urban terrain is needed for comparison (e.g. facades, rooftops). In our experiments, even this information originated from ALS data, which were recorded under optimal DGPS conditions, but it might as well be derived from any other existing 3D city model.

Within our approach, we assume that INS navigation is continuously available and that we have an initial guess of the sensor position and orientation ( $\pm 50$  m,  $\pm 5^\circ$ ). If we have to

navigate through GPS dropouts, the positioning accuracy will degrade because of INS drift effects, but we can assume that the measured ALS data are still roughly aligned to the stored information. In addition to their position in 3D space, features like size and normal direction are assigned to all segmented planar patches, thus it is comparatively easy to find corresponding objects in the database. We use this information to achieve precise alignment between the measured ALS data and the existing city model, which finally enables us to correct the presumed sensor position.

### 1.3 Related work

In recent years, airborne laser scanning systems have been explored by various scientists from different points of view. The complexity of ALS data acquisition leads to a number of potential error sources. Schenk (2001) and Filin (2003) address this problem and categorize different influences that should be considered. In addition to varying exactness of the navigational information sources, several systematic effects can lead to reduced point positioning accuracy. Exemplary limiting factors are scanning precision and range resolution of the specific laser scanning device. Other negative effects can be introduced by inaccurate synchronization of the system components. Considerable discrepancies are caused by mounting errors or disregarded lever arms (displacements between laser scanner, INS, and GPS antenna). Skaloud and Lichti (2006) approached this problem with a rigorous method to estimate the system calibration parameters such that 3D points representing a plane are conditioned to show best possible planarity. In order to use ALS within the scope of aircraft navigation, we suppose that the sensor system has been calibrated beforehand.

Some procedures described in this paper are concerned with the segmentation of point clouds into planar surfaces. Many different methods regarding this topic can be found in literature. Some authors are interested in detecting even more kinds of objects like spheres, cylinders, or cones. Rabbani et al. (2007) describe two methods for registration of point clouds, in which they fit models to the data by analyzing least squares quality measures. Vosselman et al. (2004) use a 3D Hough transform to recognize structures in point clouds. Filin & Pfeifer (2006) propose a segmentation method that is based on cluster analysis in a feature space. The RANSAC algorithm (Fischler & Bolles, 1981) has several advantages to utilize in the shape extraction problem (Schnabel et al., 2006). We apply a RANSAC-based robust estimation technique to fit straight line segments to the scan line data. In a similar manner, a modified version of this method is used to identify locally planar patches in the model data. The amount of outliers lets us distinguish between buildings and irregularly shaped objects like trees. Fundamental ideas on fast segmentation of range data into planar regions based on scan line analysis have been published by Jiang and Bunke (1994). Their algorithm divides each row of a range image into straight line segments which are combined in a region growing process. Despite the fact that we are considering continuously recorded scan lines instead of range images, we basically follow this approach during the on-line data analysis.

Several existing concepts of terrain-based navigation for aerial vehicles can be found, e.g. image based navigation (IBN), terrain-following radar (TFR), or terrain contour matching (TERCOM). Other than these methods, laser scanning is a comparatively new technique. Toth et al. (2008) propose the use of LiDAR for terrain navigation, as it provides distinct 3D measurements that can easily be used for exact comparison to

previously recorded data. In their concept, the iterative-closest-point algorithm (Besl & McKay, 1992) is chosen for surface matching. Instead of an ICP approach, we identify matching planar objects with regard to several geometric features (i.e. position, principal components, normal direction). Similar methods have demonstrated high performance for markerless TLS registration (Brenner et al., 2008). The problem of determining the transformation parameters is transferred to a system of linear equations that can be solved immediately.

## 2. EXPERIMENTAL SETUP

Data used for this study were collected during field campaigns in 2008 and 2009, using the equipment that is briefly described in this section. A more thorough description of the sensor system can be found in (Schatz, 2008).

### 2.1 Sensor carrier

The sensors described below have been attached to a helicopter of type Bell UH-1D (Figure 1). Laser scanner and IMU are mounted on a common sensor platform at the side of the helicopter, which can be tilted to allow different perspectives, i.e. nadir or oblique view. In an operational system, the pilot must be able to react to upcoming dangers, e.g. during degraded visibility conditions. Therefore, an obliquely forward-looking sensor configuration was used in our experiments. The lever arms of the components in the system are known, and the correct bore-sight angles have been determined. Calibration of these parameters is not topic of this paper, suitable methods can be found in (Skaloud & Lichti, 2006).

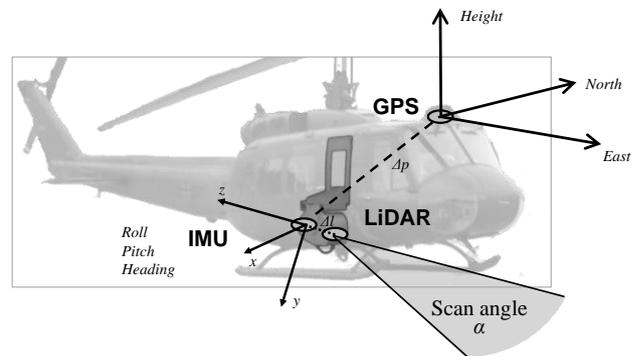


Figure 1. Sensor carrier and sensor configuration.

### 2.2 Laser Scanner

The RIEGL LMS-Q560 (2006) laser scanner makes use of the time-of-flight distance measurement principle with a pulse repetition rate of 100 kHz. Opto-mechanical beam scanning provides single scan lines, where each measured distance can be georeferenced according to position and orientation of the sensor. Waveform analysis can contribute intensity and pulse-width as additional features, but since we are mainly interested in fast acquisition and on-line processing of range measurements, we neglect full waveform analysis throughout this paper. Range  $d$  under scan angle  $\alpha$  (Figure 1) is estimated corresponding to the first significant echo pulse as it can be found by constant fraction discrimination. Typically, each scan line covers a field of view of  $60^\circ$  with 1000 range measurements ( $d, \alpha$ ) that can be converted to 2D Cartesian coordinates (Figure 6). Navigational data are synchronously assigned to these range measurements for direct georeferencing.

### 2.3 Navigational sensor system

The Applanix POS AV 410 comprises a GPS receiver and a gyro-based inertial measurement unit (IMU), which is the core element of the inertial navigation system (INS). The GPS data are used for drift compensation and absolute georeferencing, whereas the IMU determines accelerations with high precision. These data are transferred to the position and orientation computing system (PCS), where they are fused by a Kalman filter, resulting in position and orientation estimates for the sensor platform. In addition to the real-time navigation solution, specialized software can be used for accurate postprocessing of the recorded navigational data. Applanix POSpac MMS incorporates the use of multiple DGPS reference stations and the import of precise GPS ephemeris information. We consider this corrected navigation solution while generating an optimal database of the urban terrain.

### 3. USED METHODS AND DATA PROCESSING

In this chapter, we distinguish two different operating modes of ALS data acquisition and processing. First, we assume that we have optimal settings for creation of an adequate database: the relevant urban area can be scanned several times from multiple aspects with a calibrated sensor, and data can be processed and optimized off-line. During this stage, we can resort to own differential GPS base stations or use according information, e.g. provided by the ‘‘Satellite Positioning Service of the German State Survey’’ (SAPOS). Under these conditions, the absolute measurement accuracy of an ALS system is typically in the order of one decimeter (Rieger, 2008).

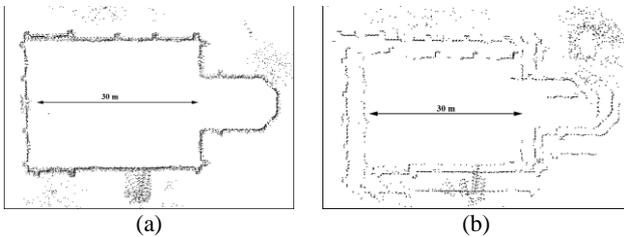


Figure 2. Horizontal cross-section of a building in overlapping point clouds: (a) after INS/DGPS postprocessing, (b) using the real-time navigation solution.

In the second mode of ALS operation, the data is used for on-line navigation updates during helicopter missions. At this time, we expect non-differential GPS conditions, GPS dropouts, and loss of data points due to smoke, fog, or other negative influences. Figure 2 shows the accuracy that can be obtained in the different operating modes. ALS data in this example were acquired at a skew angle of 45 degree (forward-looking). The helicopter approached the same urban area from six different directions, and the resulting 3D points were combined into a single point cloud. Both illustrations depict the aggregated data within the horizontal cross-section of a building in the overlap area. Best accuracy as shown in Figure 2a results from a global optimization of the navigational data with the Applanix postprocessing software POSpac. In this example, the data of six DGPS ground stations were taken into account. Compared with this accuracy, discrepancies of several meters can occur if the real-time navigation solution is used (Figure 2b). This situation will even get worse in case of GPS dropouts, depending on the quality of the IMU.

### 3.1 Automatic generation of an adequate database

The intended utilization of LiDAR sensors for aircraft guidance does not require a highly detailed GIS. We limit the creation of a database to the extraction of planar patches in multi-aspect ALS point clouds of the relevant urban area. As mentioned before, these data should be collected under optimal conditions (Figure 2a). The combined complete 3D point cloud contains information concerning all facades and rooftops of buildings. A workflow of off-line processing methods is used to filter points and extract most of the planar objects. The respective flowchart is illustrated in Figure 3.

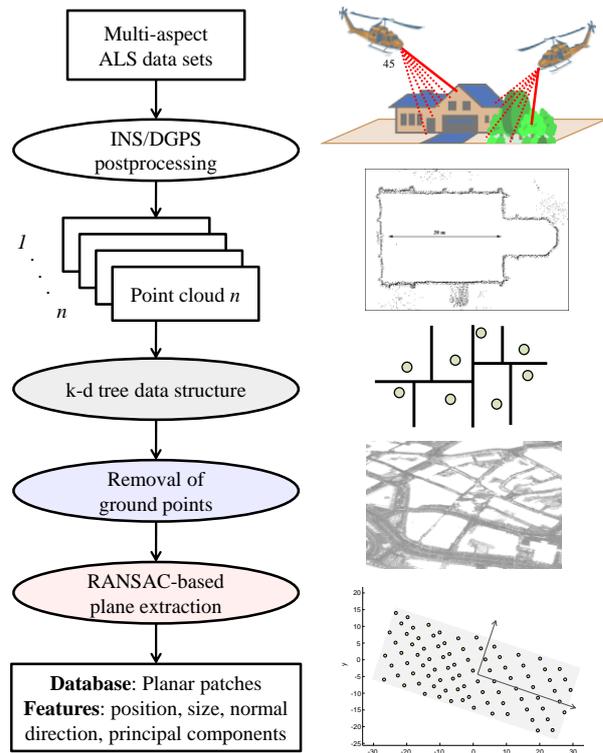


Figure 3. Flowchart of the model creation.

Merging of several multi-aspect ALS data sets results in an irregularly distributed 3D point cloud. We introduce a k-d tree data structure to handle automatic processing of these data. The search for nearest neighbors can be done very efficiently by using the tree properties to quickly eliminate large portions of the search space.

The subsequent segmentation method is intended to keep only those points that are most promising to represent parts of buildings. At first, we remove all ground points by applying a region growing technique in combination with a local analysis of height values. We search for sections of the point cloud in which the histogram of height values clearly shows a multimodal distribution. There, laser points at ground level appear as the lowest distinct peak. Such positions are then used as seed points for the region growing procedure, which collects all points falling below a certain slope. The necessary search operations are accomplished by means of the k-d tree data structure. In general, this method may misclassify some points (e.g. inner courtyards), but this is negligible for our application. An overview of advanced methods for bare-earth extraction can be found in (Sithole & Vosselman, 2004).

The main step of the model creation is the extraction of planar features from the remaining 3D points. Remarkably, the applied segmentation method is almost identical to the algorithm used for detection of straight line segments in the 2D scan line data, except for the terms line/plane and the different data structure. Similarly, geometric features of the extracted shapes have to be computed in both the model and the on-line results. These topics are described in sections 3.2 and 3.4, respectively. Figure 4 gives an impression of the derived model data. The underlying point cloud is composed of four partial scans of the terrain (different aspects).



Figure 4. Partial view of the database with ground level (blue), vegetation (green) and planar patches (red).

### 3.2 Scan line analysis of airborne LiDAR data

During on-line processing, the analysis of geometric features is performed directly on the scan line data. Most parts of typical buildings will appear as local straight line segments in the 2D Cartesian representation, even if the airborne laser scanner is used in oblique configuration (Figure 1). The RANSAC technique is used to locally fit straight line segments to the scan line data. As mentioned in section 3.1, the algorithm described below can be modified to accomplish segmentation of planar surfaces in a 3D point cloud. This is simply done by replacing the scan line index with the k-d tree data structure, and by turning attention to 3D planes instead of 2D lines.

An overview of the proposed method is shown in Figure 5. Within each new scan line, the 2D points (with range and scan angle converted to Cartesian coordinates) are first evaluated by a local principal component analysis. This step lets us distinguish between regularly or irregularly distributed point clusters. Within each following iteration step, we randomly select scan line positions of regularly distributed data points and try to fit a straight line segment to the neighboring data. The RANSAC technique provides a robust estimation of the line segment's parameters. If the fitted straight line is of poor quality, the data associated with the current position is assessed as clutter. Otherwise, we try to optimize the line fitting by looking for all data points that support the previously obtained line, which is done in steps (10), (11), and (12). These steps actually represent a "line growing" algorithm. The local fitting of a straight line segment is repeated with the supporting points to get a more accurate result. The end points of each line segment can be found as the perpendicular feet of the two outermost inliers. Figure 6 shows detected straight lines for an exemplary scan line, depicted with a suitable color-coding according to the normal direction.

	<ol style="list-style-type: none"> <li>(1) Perform a local principal component analysis for the 2D data points in the new scan line <math>A</math>. The respective smallest eigenvalue indicates the local straightness of the data.</li> <li>(2) Choose an unprocessed position <math>i</math> among the available data in the array <math>A</math> where data points were found to be regularly distributed.</li> <li>(3) Check a sufficiently large interval around this position <math>i</math> for available data, resulting in a set <math>S</math> of 2D points.</li> <li>(4) Set the counter <math>k</math> to zero.</li> </ol>
	<ol style="list-style-type: none"> <li>(5) If <math>S</math> contains more than a specific number of points, go to (6). Otherwise mark this position <math>i</math> as discarded and go to step (15).</li> <li>(6) Increase the counter <math>k</math> by one.</li> <li>(7) Perform a RANSAC-based straight line fitting with the 2D points in the specified set <math>S</math>.</li> <li>(8) If the number of inliers is low, mark the current position <math>i</math> as discarded and go to step (15).</li> <li>(9) Obtain the Hessian normal form <math>L: (x-p) \cdot n_0 = 0</math> and push the current position <math>i</math> on a stack (LIFO).</li> </ol>
Region growing	<ol style="list-style-type: none"> <li>(10) Pop the first element <math>j</math> off the stack.</li> <li>(11) Check each position in an interval around <math>j</math>, which has not already been looked at, whether the respective point lies sufficiently near to the straight line <math>L</math>. If so, include the 2D point in a new set <math>S'</math>. Additionally push its position on the stack if indicated by (1).</li> <li>(12) While the stack is not empty, go to (10). Otherwise continue with step (13).</li> </ol>
	<ol style="list-style-type: none"> <li>(13) If the counter <math>k</math> has reached its predefined maximum (e.g. two cycles), mark all positions of points in <math>S'</math> as processed and determine the regression line to <math>S'</math>. Store the perpendicular feet of the two outermost points to define the straight line segment and go to step (15). Otherwise continue with (14).</li> <li>(14) Go to step (5) with the new set of points <math>S := S'</math>.</li> </ol>
	<ol style="list-style-type: none"> <li>(15) Repeat from (2) until all points are classified.</li> </ol>

Figure 5. Procedure for RANSAC-based shape extraction (example: detection of lines in a set of 2D points).

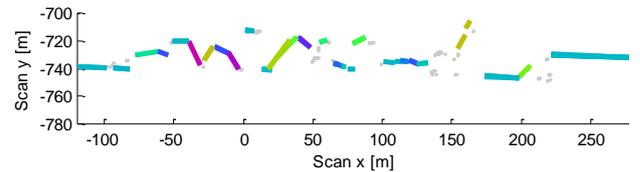


Figure 6. Detected straight line segments in a typical scan line.

### 3.3 Grouping of line segments

The end points of each line segment are georeferenced to result in correct positioned straight 3D lines. In this section, we describe a procedure to connect coplanar line segments of consecutive scan lines. Let  $P_i$ ,  $P_j$ , and  $P_k$  be three of the four end points of two line segments in different scan lines. The distance of the fourth end point  $P_m$  to the plane defined by the three others is a measure of coplanarity. We define a distance  $d_p$  as the sum of all four possible combinations:

$$d_p := \sum_4 \frac{|(P_i - P_m)^T (P_i - P_j) \times (P_i - P_k)|}{\|(P_i - P_j) \times (P_i - P_k)\|} \quad (3.1)$$

The algorithm to find corresponding line segments in a sequence of scan lines can be summarized as follows:

- (1) Select the next line segment  $a$  in the current scan line.
- (2) Set the label of  $a$  to a new and increasing labeling number.
- (3) Successively compare line segment  $a$  to each line segment  $b$  of several previous scan lines. If Euclidean distances, disparity of normal direction, and the measure of coplanarity  $d_p$  are found to be smaller than predefined thresholds, go to step (4). Otherwise go to step (5).
- (4) Set the label of  $a$  to that of  $b$ .
- (5) Continue with (1) until all line segments  $a$  are processed.

The above steps summarize the main ideas of our method. In fact, we apply an extended two-pass approach to improve detection of connected components. More details on this topic can be found in (Hebel & Stilla, 2008). Figure 7 illustrates the procedure. First, each line segment is initialized with a unique label. Coplanar line segments that are found to lie near to each other are linked together by labeling them with a common labeling number. This process is repeated until all new line segments are labeled. Surfaces are represented by the emerging clusters of line segments with the same label (Figure 8).

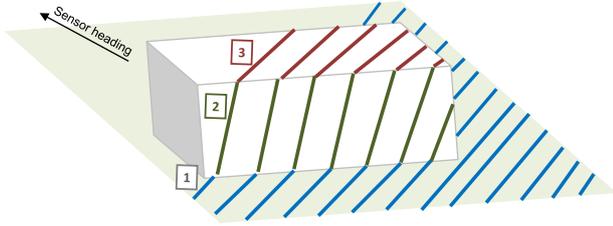


Figure 7. Illustration of scan line grouping.



Figure 8. Result of scan line grouping for ALS data (5 seconds).

### 3.4 Feature extraction

Each cluster of connected straight line segments can be characterized by a set of features which are described in this section. For a given cluster of connected line segments, let  $C$  denote the set of associated 3D data points. The centroid of  $C$  can be computed as the sum of all points divided by their number, and  $C$  can be translated towards the origin:

$$\bar{c} = \frac{1}{n} \sum_{i=1}^n c_i, \quad C_o := C - \bar{c} \quad (3.2)$$

The eigenvectors of the covariance matrix  $C_o^T C_o$  are the principal components of  $C$ . The normal direction  $\mathbf{n}_o$  is given as the normalized eigenvector that corresponds to the smallest

eigenvalue. The value of the smallest eigenvalue  $\lambda_o$  of the covariance matrix, divided by the number of points, is influenced by the curvature and the scatter of  $C$ . If it is near zero, this indicates a planar surface. The features used to identify matching surfaces in the model data and the results of scan line analysis are: centroid, normal direction, and the normalized eigenvalues of the covariance matrix. These features can even be used to classify and remove irregularly shaped surfaces, e.g. the ground level in Figure 8.

### 3.5 Registration of ALS and model data

Even without considering terrain-based navigation, we assume that the sensor position is known approximately e.g. up to 50 meter. In case of GPS dropouts, the IMU drift will not distort the positioning exactness dramatically. The relative accuracy provided by the INS measurements still ensures consistent ALS measurements over limited periods of time, depending on the quality of the INS system. In some situations, the absolute navigational accuracy needs to be improved. Examples are low-altitude flights of helicopters at night or preparation of landing approaches during rescue missions at urban areas.

If the helicopter is equipped with a LiDAR sensor, ALS data can be collected for several seconds in order to scan the urban area in front of the helicopter (Figure 8). Surfaces that are instantaneously detected in these data can be compared and matched to the existing database of the terrain (Figure 4). The features that are used to establish links have been described in section 3.4. First, the displacement of the centroids has to fall below a maximum distance. Second, the angle between the normal directions should be small (e.g.  $<10^\circ$ ). Third, the normalized eigenvalues of the covariance matrix  $C_o^T C_o$  should be similar. Nevertheless, wrong assignments may occur especially when considering a large search radius. In order to be robust against these perturbations, the procedure described in this section is complemented by a RANSAC scheme. Figure 9 illustrates an exemplary pair of associated surfaces. The offset in position and orientation indicates the inaccuracy of the navigational data.

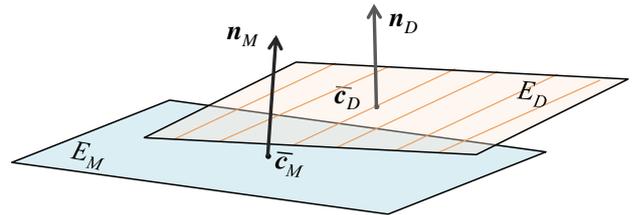


Figure 9. A pair of corresponding planes in model  $M$  and currently acquired ALS data  $D$ .

In this section, we determine a rigid transformation  $(R, \mathbf{t})$  to correct these discrepancies. Let  $E_M$  denote the planar surface of the model  $M$  that is associated with the plane  $E_D$  in the currently collected ALS data  $D$ . The respective Hessian normal form of these planes is given by the centroids and the normal directions  $\mathbf{n}_M, \mathbf{n}_D$  (Figure 9). Since both planes should be identical after registration, the centroid of  $E_D$  should have zero distance to  $E_M$ . Moreover, the two normal vectors should be equivalent if they are normalized to the same half space. In addition to these conditions, we can assume that errors of orientation will not exceed the range of  $\pm 5^\circ$ . That enables us to linearize the equations:

$$\begin{cases} (R \cdot \bar{c}_D + t - \bar{c}_M) \cdot n_M = 0 \\ (R \cdot n_D) \cdot n_M = 1 \end{cases}, \quad R \approx \begin{pmatrix} 1 & -\alpha_3 & \alpha_2 \\ \alpha_3 & 1 & -\alpha_1 \\ -\alpha_2 & \alpha_1 & 1 \end{pmatrix} \quad (3.3)$$

The rotation angles ( $\alpha_1, \alpha_2, \alpha_3$ ) and the translation components ( $t_1, t_2, t_3$ ) are the six variables to be determined. Each corresponding pair of planes  $E_D, E_M$  yields two linear equations (3.3), therefore at least three pairs have to be identified in the data to compute the rigid transformation ( $R, t$ ). In general, more correspondences can be found at urban areas. The resulting overdetermined system can be solved approximately by inverting the normal equations. In addition, the area of the planar patches can be used as a weighting factor. Finally, the corrected position of the sensor in the model coordinate system is given as  $R \cdot p_{GPS} + t$  and the orientation is corrected to  $R \cdot R_{IMU}$ .

#### 4. EXPERIMENTS

We tested the proposed methods on the basis of real sensor data which were recorded 300 meters above Abenberg, Germany. Data available from four flights over this urban terrain in 2008 led to the database shown in Figure 4. Data collected in 2009 were considered to prove the concept of terrain based navigation (Figure 8). For this purpose, 8000 randomly chosen displacement vectors in the range [5 m, 50 m] were added to the exact sensor position and it has been checked if these offsets are corrected automatically. Figure 10 shows the average displacement between calculated and exact sensor position against the number of matching pairs of planes. With our data, we were able to reduce the average offset in sensor position to 0.6 m if at least 10 pairs of associated surfaces can be found (standard deviation: 0.2 m). When considering a larger number of matching planes (50), this result was even improved to 0.4 m. These numbers most likely depend on additional conditions, e.g. aircraft altitude, aircraft speed, number and orientation of facades and rooftops.

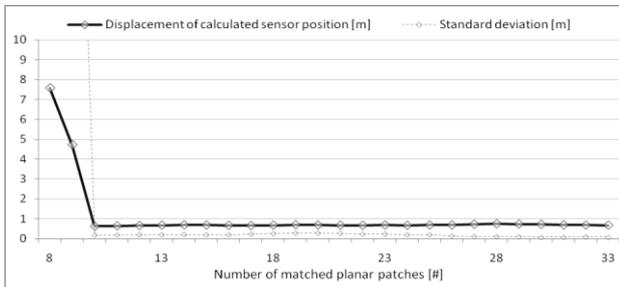


Figure 10. Average displacement against number of planes.

#### 5. CONCLUSION AND FUTURE WORK

The examples presented in this paper were obtained with an experimental sensor system, for which data analysis can only be done offline to show the feasibility of the proposed approach. Nevertheless, we guess that all computations can be accomplished in real-time, with an efficient implementation and appropriate hardware. In our experiments, we were able to align the model and the ALS data such that matching objects show an average distance of 8 cm after the registration. This absolute exactness is not necessarily transferable to the sensor position (see Section 4). With a larger distance between helicopter and the terrain, impreciseness of the sensor orientation has a considerably higher impact on the overall displacement. For example, an angular error of  $0.1^\circ$  would lead to a shift of 1 m in

a distance of 600 m. The absolute exactness of the estimated sensor position improves significantly when considering larger areas and/or shorter ranges, e.g. when approaching the terrain at low altitude. In future work, we will analyze these influences in more detail, and we will focus on on-line change detection.

#### 6. REFERENCES

- Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, pp. 239-256.
- Brenner, C., Dold, C., Ripperda, N., 2008. Coarse orientation of terrestrial laser scans in urban environments. *ISPRS Journal of Photogrammetry and Remote Sensing* 63 (1), pp. 4-18.
- Filin, S., 2003. Recovery of Systematic Biases in Laser Altimetry Data Using Natural Surfaces. *Photogrammetric Engineering & Remote Sensing* 69 (11), pp. 1235-1242.
- Filin, S., Pfeifer, N., 2006. Segmentation of airborne laser scanning data using a slope adaptive neighborhood. *ISPRS Journal of Photogrammetry and Remote Sensing* 60 (2), pp. 71-80.
- Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *CACM* 24 (6), pp. 381-395.
- Hebel, M., Stilla, U., 2008. Pre-classification of points and segmentation of urban objects by scan line analysis of airborne LiDAR data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 37, Part B3a, pp. 105-110.
- Jiang, X., Bunke, H., 1994. Fast Segmentation of Range Images into Planar Regions by Scan Line Grouping. *Machine Vision and Applications* 7 (2), pp. 115-122.
- Rabbani, T., Dijkman, S., van den Heuvel, F., Vosselman, G., 2007. An integrated approach for modelling and global registration of point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing* 61 (6), pp. 355-370.
- Rieger, P., 2008. The Vienna laser scanning survey. *GEOconnexion International Magazine*, May 2008, pp. 40-41.
- Schatz, V., 2008. Synchronised data acquisition for sensor data fusion in airborne surveying. *Proceedings of the 11th International Conference on Information Fusion*, 1-6.
- Schenk, T., 2001. Modeling and Analyzing Systematic Errors in Airborne Laser Scanners. *Technical Notes in Photogrammetry* 19. The Ohio State University, Columbus, USA. 42 p.
- Schnabel, R., Wahl, R., Klein, R., 2006. Shape Detection in Point Clouds. *Technical report No. CG-2006-2*, Universitaet Bonn, ISSN 1610-8892.
- Schulz, K.R., Scherbarth, S., Fabry, U., 2002. HELLAS: Obstacle warning system for helicopters. *Laser Radar Technology and Applications VII, Proceedings of the International Society for Optical Engineering* 4723, pp. 1-8.
- Sithole, G., Vosselman, G., 2004. Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing* 59 (1-2), pp. 85-101.
- Skaloud, J., Lichti, D., 2006. Rigorous approach to bore-sight self-calibration in airborne laser scanning. *ISPRS Journal of Photogrammetry & Remote Sensing* 61 (1), pp. 47-59.
- Toth, C.K., Grejner-Brzezinska, D.A., Lee, Y.-J., 2008. Recovery of sensor platform trajectory from LiDAR data using reference surfaces. *Proceedings of the 13th FIG Symposium and the 4th IAG Symposium*, Lisbon, Portugal, 10 p.
- Vosselman, G., Gorte, B.G.H., Sithole, G., Rabbani, T., 2004. Recognising structure in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* 46 (8), pp. 33-38.