Plane Based Free Stationing for Building Models

Christian MANTHE, Germany

Key words: plane based building model, plane detection, plane based transformation

SUMMARY

3D Building models are used to construct, manage and rebuild buildings. Thus, associated issues are: How accurate has an object been built compared to its plan or model? How accurate can we determine the surface of single entities of buildings? Is the given model information sufficient to plan modifications?

It is shown how planes in a model can be used for free stationing, inspecting and updating a model. A plane based model for data storage is presented, which enables the possibility of updating the building model during its live time. Hence this approach can be used to navigate within the model and furthermore document, control and update the dataset.

Based on SketchUp plug-ins, individual steps for updating a model are explained. These are as follows: extraction of plane parameters from point clouds, determination of initial transformation parameters, plane based matching (measurements vs. model) and finally adaptation of the model. It is shown how the presented method is usable for datasets from different measurement systems.

Plane Based Free Stationing for Building Models

Christian MANTHE, Germany

1. INTRODUCTION

If a building has been designed by an architect the 3D model is mostly given. In that case it can be used to create a 3D facility management system that has been derived from the given model. Based on that 3D model new possibilities can be explored in marketing, transparent management, navigation or planning of reconstruction processes. The task for surveyors is to determine deviations or the gap between the model and the real world with respect to a desired accuracy.

The situation in the as-built segment is even more difficult. Of course, these facilities are managed without geometric models, but nevertheless it would be helpful to use only one dataset for all mentioned applications.

When creating a 3D model of a building, different demands concerning its topology and geometry to create a useful model have to be satisfied. A useful model on one hand describes a dataset that allows the integration of new observations or where given model can be updated in terms of its geometry. On the other hand the model should use the topology of rooms as an interface to facility management software.

2. A PLANE BASED BUILDING MODEL

For the creation of a 3D model distance measurement devices can be used to virtually construct a building within CAD software. The derived 3D model does only contain geometric information. Stochastic information of the model can only be assumed globally with respect to the applied measurement devices but is only sufficient to answer simple tasks.

The problem of this naive perspective is that the local precision is influenced by the order of how the model has been created which also falsifies the accuracy of the local geometries as well as the stochastic information. The only current solution of how to check the derived CAD model is by carrying out control measurements.

The best way to receive stochastic information is by performing an adjustment where all measurements are used. The advantage of the adjustment is that the solution is independent of the order of the construction process of the CAD which opens the possibility to update the model geometry with new observations. To design functional relations identifiers for all entities have to be introduced to handle the redundant observations.

By creating the functional model suitable unknowns has to be chosen. Because of the reduced number of unknowns and the characteristics of buildings, plane parameters were introduced to describe the functional model (Clemen, 2009). This approach separates geometric information (plane parameters) from the topology (vertices, edges and faces), which allows updating processes of the geometry as a result of the adjustment.

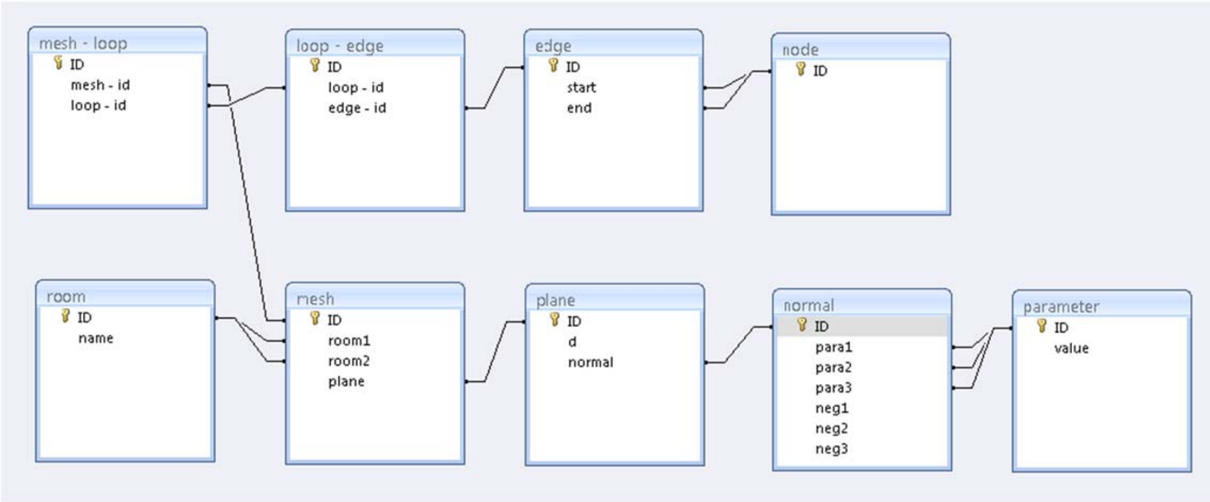

Figure 1: entity relationship model for a plane based building model

The used entity relationship model (ERM), as shown in Figure 1, shows that nodes only contain identifiers as attributes but no coordinates. The coordinates of the nodes have to be calculated from plane intersections. That leads to a validation law of a given model. It means every node or point in the model has to be connected to more than 2 planes with different normal vectors. The first step to handle and validate a building model is to make sure that the rooms or spaces in the whole building are closed which also describes an interface to facility management software. Therefore the room aggregation plug-in for SketchUp like described in (Manthe 2010) is used.

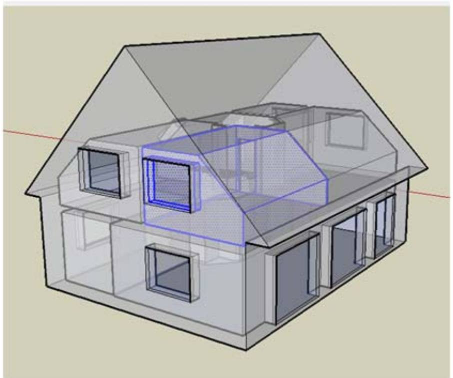

Figure 2: building as coposite of rooms

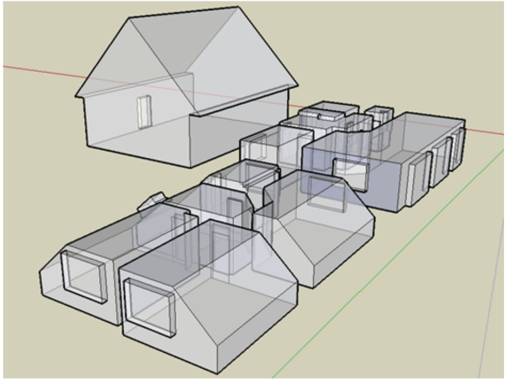

Figure 3: rooms of an building

In the second step all nodes have to be controlled with respect to the related planes or faces. After that validation process the building model can be converted into the plane based data structure.

3. PLANE SETS TO MATCH OBSERVATIONS WITH THE MODEL

To increase the precision of the model redundant observations have to be stored within the data model with respect to its given topology. Therefore we have to find the related identifier for each observation, which is a difficult task in three dimensions. For every distance measurement the corresponding entity identifier has to be manually found to store the observation related to the data model.

The use of plane sets derived from point clouds enables the possibility of applying semi automatic observation matching. By using plane sets the user has to find only three or four homologue planes to calculate approximated transformation parameters. After the transformation of the plane set into the model coordinate system other corresponding planes can be detected automatically.

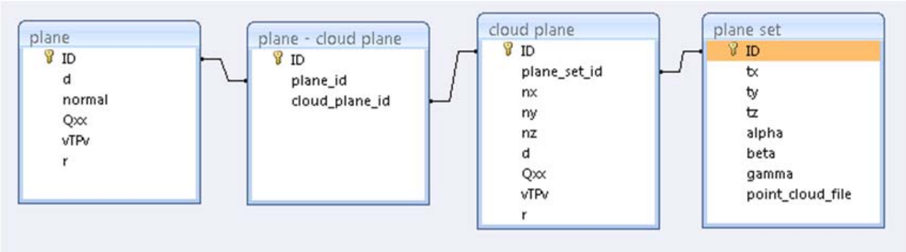

Figure 4: extended ERM with respect to observed plane sets

That leads to an extension of the data model with respect to the observed planes shown in Figure 4. For every model plane we have to find the corresponding plane within the point cloud. After at least three homologous planes have been found its transformation can be calculated which are equal for all planes within a plane set. Therefore we have to extract planes automatically from the point cloud for the matching process.

3.1 Plane Detection

There are different ways to extract planes from a point cloud. Generally three different situations can be found. Firstly the allocation between the points and its corresponding plane are given. This could have been achieved by a terrestrial measurement with a total station. Secondly just the neighborhood of the points is given via the measurement procedure e.g. laser scanning. This approach is used to detect planes automatically in (Gielsdorf, 2009) and (Dold, 2010). Thirdly the neighborhood of the points is not given. For example the point cloud was created based on photgrammetry or registered laser scans.

The third case was chosen as it is independent of the measurement technique. Based on the Random Sample Consensus (RANSAC) paradigm introduced by (Fischler & Bolles, 1981) a plane detection algorithm has been implemented. Some advantages of this approach are its ability to ignore outliers without explicit handling and the fact that it can be used to extract multiple instances in a data set (Wahl *et al*., 2005).

As described in (Wahl *et al*., 2005) the number of necessary iterations l to find a plane in a dataset depends logarithmically on the inverse failure rate p_f and quadratic to the number of planes k, assuming that the number of the existing planes with in the dataset is much lower than the number of points n.

$$l = \ln(1/p_f)k^2$$

Usually the number of existing planes k is unknown in arbitrary point clouds. In order to solve this problem the given point cloud has to be segmented into regions until the number of planes can be determined. That is possible by splitting the point cloud recursively in 8 cubed subspaces until the maximal number of contained points is reached. Therefore we create a space index based on kd-tree(Wikipedia) with a spatial search function (Manthe 2006). As a result a tree structure has been determined where the size to the cubed subspaces is depended onto the distribution of the points in the point cloud. In this tree structure all leafs are regions with the same number of points where no more than two planes are detectible. In those local regions we try to find the local planes with $l = \ln(1/p_f)2^2$ tries were p_f can be chosen manually. By $p_f = 0.01\%$ the number of useful tries to detect a plane is $\ln(100) * 4 \approx 18$ times.

After l tries different planes were detected. The best plane is represented by the most points with respect to a given threshold value (distance to the plane). To get stochastic features of the best plane an adjustment with the received points is carried out. Therefore a parametric adjustment is used which is constrained by the normal equation ($n_x^2 + n_y^2 + n_z^2 = 1$).

The received points will be removed from the original dataset before the algorithm starts tries to find the next plane. In order to find the best description of a plane all planes from adjacent sub entities within the tree structure have to be unified in case that their plane parameters are similar. If this is the case then the segmented points from all sub planes are processes within another adjustment which can be found within the next chapter.

3.2 Merging the local planes to a plane set

I order to merge plane pairs of adjacent cubes the weighted arithmetic mean of the standard deviations of the unit weight s_{0m} of both planes has to be compared to the standard deviation of the unit weight s_0 from an adjustment of the plane based on both point sets.

$$H0: \quad F = \frac{{s_{0m}}^2}{{s_0}^2} < F_s(r_m, r, 95\%)$$

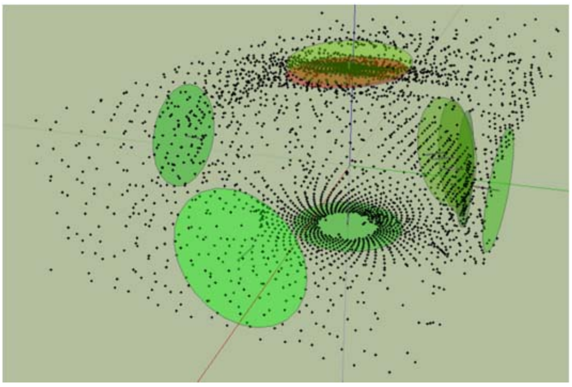

Figure 5: plane set out of a point cloud

With respect to the given space index structure the planes will be linked together in order to receive stochastic information. Finally a plane set with stochastic descriptions as features of a point cloud has been derived as shown in Figure 5. To estimate the transformation parameters homologous planes have to be found in a matching process.

4. PLANE BASED MATCHING

In order to add 3D information from a point cloud to a building model transformation parameters are needed. To determine these parameters homologous planes have to be used instead of points. The disadvantage of homologous points is its visibility during the lifetime of the building. It is easier to find points at the same plane then just discrete points. The biggest advantage by using planes for matching purposes is that planes can be detected automatically with its stochastic information.

4.1 Functional Model

To find transformation parameters from a given plane a functional relationship between plane parameters and the searched transformation parameters have to be determined. Figure 6 displays two planes where plane 1 is a model plane whereas plane 2 is a plane from a point cloud. Both planes corresponding to each other thus they describe the same face in real world.

Both planes can be described with the Hessian Normal form (1) and (2).

$$0 = \vec{n}_1 \vec{x}_{p1} - d_1 \quad (1) \qquad\qquad 0 = \vec{n}_2 \vec{x}_{p2} - d_2 \quad (2)$$

The equations above are true if the points $\vec{x}_p$ are located on a plane.

The foot point of plane1 can be described as shown in equation (3) by the normal $\vec{n}_1$ and the distance parameter d_1.

$$\vec{x}_F = \vec{n}_1 d_1 \quad (3)$$

If the foot point will be transformed by applying the searched transformation parameters between both planes it fulfills the hessian normal equation of the second plane (2).

$$\vec{x}_{p2} = R\vec{x}_{foot\_p1} + \vec{t} \quad (4)$$

Equation (4) shows the point transformation between two systems where R is the rotation matrix and $\vec{t}$ is the vector for the translations. By substituting (4) in equation (2) and the use of the foot point equation (3) we get finally (5).

$$0 = \vec{n}_2 \left(R\vec{x}_{foot_{p1}} + \vec{t} \right) - d_2$$

$$0 = \vec{n}_2 R\vec{x}_{foot\_p1} + \vec{n}_2 \vec{t} - d_2$$

$$0 = \vec{n}_2 R\vec{n}_1 d_1 \; + \vec{n}_2 \vec{t} - d_2 \quad (5)$$

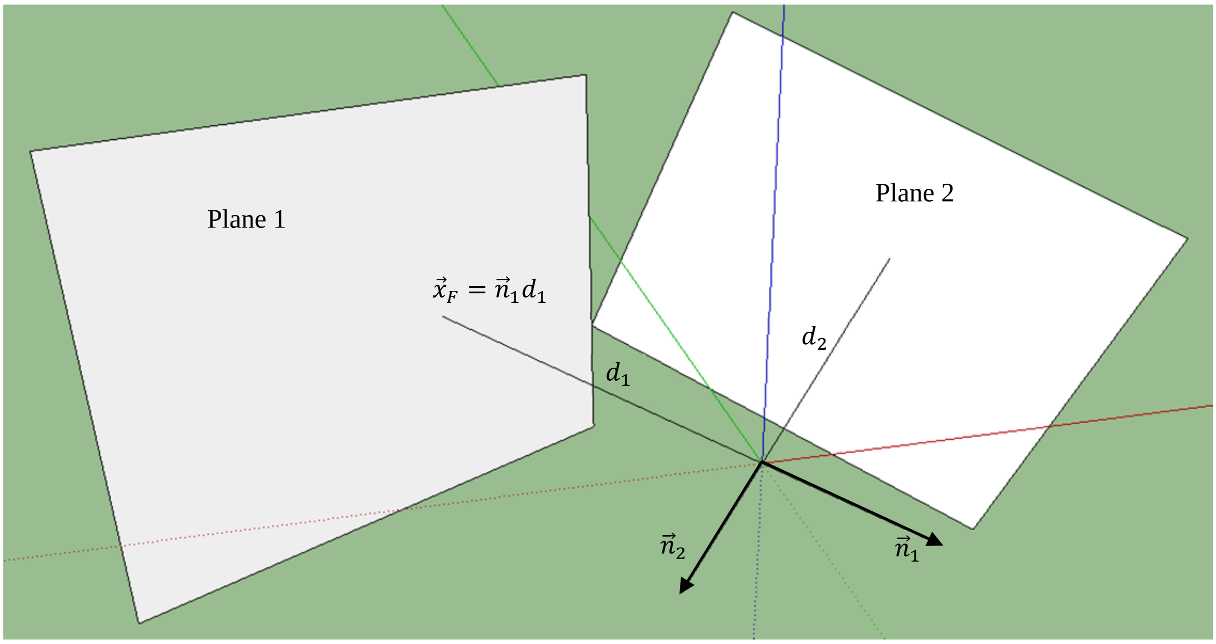

Figure 6: two homologous planes from different plane sets

In order to find an independent functional relationship between the normalized plane vectors with respect to the rotation matrix equation (6) is introduced.

$$\vec{n}_2 = R\vec{n}_1 \quad (6)$$

It shows the functional relation to estimate rotation parameters from two planes. For three unknowns three equations are needed. If more homologous planes are used more equations have to be introduced which leads to a conditional adjustment with unknown parameters. By using three plane pairs nine equations with three unknowns arise. In order of being independent from the approximated rotation parameters quaternions were used as unknowns. By substituting (6) in (5) equation (7) can be derived were $\vec{n}_2\vec{n}_2 = 1$ because of the constraint that the length has to satisfy $|\vec{n}_2| = 1$. Finally a functional relationship between our known plane parameters and the searched translation parameters in (8) has been derived.

$$0 = \vec{n}_2\vec{n}_2 d_1 + \vec{n}_2\vec{t} - d_2 \quad (7)$$

$$0 = d_1 + \vec{n}_2\vec{t} - d_2 \quad (8)$$

For three unknown translations (t_x, t_y, t_z) one equation for one plane pair is given. A minimum of three plane pairs has to be found in order to estimate the translation parameters.

Finally two independent equation systems to estimate the translation and rotation parameters of a plane set have been determined. In order to solve these equations corresponding planes have to be introduced as described in the following paragraph.

4.2 Matching of corresponding planes

The aim of the matching approach in this case is to automatically find corresponding planes. By initially setting three homologous planes approximated transformation parameters can be determined.

With these transformation parameters the plane set can be transformed into the model system. A kd-tree can be used to find candidates for homologous planes by the use of a search function by using the plane parameters as sign vector attributes. To make sure to match only identical planes its stochastic information has been used. Therefore a comparison of the empiric standard deviations of the unit weight of the candidates with the F-test has been performed.

To receive better transformation parameters the new plane pairs have been added to the adjustment. This matching and estimation of the transformation parameter will be repeated until no new plane pairs were found.

At the end of the matching process the transformation parameters of the plane set and the homologous planes between the model and the plane set have been determined. As the plane sets are features extracted from a point cloud the computed transformation parameters can be used to transform the whole point cloud into the model coordinate system.

5. GEOMETRICAL ADAPTION BASED ON OBSERVATIONS

To update the geometry of a model the underlying planes have to be changed. If more than one plane set is given an adjustment has to be solved to estimate the new geometry of the model planes. In that adjustment the model planes and the plane set transformation parameters are the observed unknowns. As described in 4.1 the same functional relationship can be used.

As a result of the previous adjustment the adjusted model planes and the adjusted transformation parameters with respect to every plane set have been determined. In addition a stochastic description of the model planes is given. The geometric model can then be updated by editing the plane entities as described in Figure 1.

After a reconstruction of the model in a CAD system such as SketchUp every node receives coordinates with stochastic information. The stochastic features are computed with the law of variance propagation. In case that a node belongs to more than three planes an adjustment has to be carried out.

Based on the stochastic information at each vertex a measurement tool within a CAD can be implement that displays standard deviation information which makes the model geometry transparent with respect to tasks in the future.

Another possibility arises for the model by adding stochastic information to a model face that is related to a plane. This makes it possible to test a defined precision for a transformation with respect to the used planes, to make sure that the used model face can be used for free stationing.

6. CONCLUSIONS

It has been shown that a plane based model can be used to register point clouds after feature extraction. Therefore an arbitrary point cloud has been divided into subspaces with a nearly equal number of points. Based on the RANSAC paradigm the planes were detected and adjusted. After the detection of the planes in the subspaces they were merged based on statistical tests to a plane set.

The used method to extract planes makes it possible to integrate different point clouds from different measurement methods (e.g. laserscanning, photogrammetry). Based on the plane based model a semi-automatic plane matching for plane sets is shown. This approach makes it possible to compare models that have been derived at different epochs.

As a result a building model with different stochastic information of each model face and vertex has been derived. If the accuracy of a part of a building is not sufficient for a task the model geometry can easily be updated by adding new local observations to the model as described. After the adjustment the geometry has been updated. The automatic reconstruction of the building based on plane intersection shows the best geometric model description with respect to the given observations.

REFERENCES

Clemen, C. (2009). Ein geometrisch-topologisches Informationsmodell für die Erfassung und Validierung von flächenparametrisierten 3d-Gebäudemodellen. München: Dissertation, Verlag der Bayerischen Akademie der Wissenschaften.

Dold, C. (2010). Ebenenbasierte Verfahren für die automatische Registrierung terrestrischer. München: Dissertation, Verlag der Bayerischen Akademie der Wissenschaften.

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

Gielsdorf, F. (2009). *Ebenendetektion, Matching und verkettete Transformation von Laserscans.* Berlin - Offenbach: Allgemeine Vermessungs-Nachrichten VDE Verlag GmbH.

Manthe, C. (2006). *Matching von Vektordaten auf Grundlage mehrdimensionaler Suchbäume und mathematischer Statistik.* Berlin: Diploma thesis (unpublished), Technical University of Berlin.

Manthe, C. (2010). *Room-Element-Aggregation to Enhance the Quality of Observed 3D Building Information.* Sydney, Australia, 11-16 April 2010: FIG Congress 2010 Facing the Challenges – Building the Capacity.

Wahl, R., Guthe, M., & Klein, R. (2005). *Identifying planes in point-clouds for efficient hybrid rendering.* The 13th Pacific Conference on Computer Graphics and Applications. S. 1-8.

Wikipedia. (n.d.). *kd-tree.* Retrieved March 2011, from http://en.wikipedia.org/wiki/Kd-tree

BIOGRAPHICAL NOTES

Christian Manthe, born 1977, Graduated in 2006 as Dipl.-Ing. in Surveying at Berlin Institute of Technology, Since 2006 Assistant at the Department of Geodesy and Geoinformation Science, Berlin Institute of Technology.

CONTACTS

Christian Manthe
Chair of Geodesy and Adjustment Theory
Department of Geodesy and Geoinformation Science
Berlin Institute of Technology, Secretariat H20
Straße des 17. Juni 135
10623 Berlin
GERMANY
Tel. +49 30 314-24147
Fax + 49 30 314-21119
Email: christian.manthe@tu-berlin.de