Using Robust Methods for Automatic Extraction of Buildings

Christophe Vestri
Research and Development
ISTAR
Sophia-Antipolis, France 06905
vestri@istar.fr

Frédéric Devernay
Chir
INRIA
Sophia-Antipolis, France 06902
devernay@sophia.inria.fr

Abstract

We present a system for modeling buildings from a single correlation-based Digital Elevation Model (DEM). The model is constructed in two stages. The first stage segments the DEM into planar surface patches that describe the building. The second stage generates the final polygonal model of the building using weak geometric constraints. We use robust estimation methods at different stages of our modeling process to develop an efficient and noise-insensitive modeling system. The proposed system is fully automatic and does not use any a priori information about the shape of the buildings. We present results on isolated buildings and on a large area of the city of Berlin.

1. Introduction

Automatic extraction of descriptions of buildings in 3D is an essential task for a variety of applications such as telecommunications and urban planning. This task is difficult because of the complexity, number and diversity of 3D objects of the urban environment. Most of the modeling task is currently done manually. The cost and time involved in manual reconstruction is high and has motivated much active research on automatic 3D detection and reconstruction of buildings. In this paper, we present a system for the automatic modeling of buildings from a single range data image, a Digital Elevation Model (DEM). DEMs can come from laser altimetry or stereo-based matching of optical images. The objective is to provide digital models to assist planning for wireless networks.

We start with a dense raw DEM that was made using a correlation-based stereo method [5]. The DEM has a 50cm resolution and is made from 37.5cm resolution aerial images. This DEM is the only input of our modeling system. The global strategy of this system consists of two stages: The first stage is the segmentation of the DEM into locally planar surfaces to recover the various facets of the buildings from the raw DEM. We merge the redundant patches and select the best patches to describe the building. The second stage is the vectorization of the boundaries of each surface patch to obtain the model of the buildings as follows: we built a synthetic DEM with the selected planar patches. We extract boundaries of the different regions of this synthetic DEM to build an initial polygonal models of the buildings. Finally, a refining procedure imposes geometric constraints to regularize the model. This modeling system processes each building independently. It is restricted to the buildings with flat roofs, but it is able to model buildings of all shapes. In another report, we have validated the results for the intented application, that is simulation of coverage for planning of wireless networks.

Previous Work A variety of methods have been used for building reconstruction (see for exemple [1, 2, 3]). These methods can be divided into model-based and strategy-based approaches. The model-based approaches integrate into the model some knowledge about the 3D real world. The strategy-based approaches use a strategy to construct the model. This strategy can be grouping, matching of primitives from multiple images, or robust approximation of hypotheses extracted from a DEM. The system described here follows the second class of approaches. We propose to use several robust methods (see for exemple [7]) to solve the complex problems of our strategy.

2. Detection of buildings

We apply the whole process one building at a time. We automatically detect and extract, from the raw DEM, each building or group of adjacent buildings. First, we build a height map by subtracting the Digital Terrain Model (DTM obtained manually) from the raw DEM. Then, we extract each blob which has a sufficient size of our height map by using an arbitrary threshold (we used 6 meters) to obtain the objects above the ground, such as buildings and vegetation.

Finally, we build a local DEM for each extracted blob by masking the ground in the raw DEM. We apply a segmentation and vectorization processes to each building (in the local DEM) independently, then merge all results to obtain the final model.

3. Segmentation of the DEM

The first objective is to extract a simple and representative description of each building in the scene without any previous knowledge of their shape. By using a DEM as the initial data, this problem can be viewed as modeling a cloud of 3D noisy data. Our approach is based on the *ExSel++* framework presented in [6]. The authors define a general framework to extract parametric models from dense or sparse data. One capability of their framework is the ability to use and select multiple models to describe the data.

The DEM is a $2\frac{1}{2}$-D map. Data from this map mainly corresponds to building roofs and ground. We have choosen the planar surface patch model to describe the different parts of the buildings. We are able to describe most of the buildings of the scene with this simple model (except domes or cylindrical shapes). The segmentation process consists of three main stages that we will describe separately: a data exploration stage which generates a list of model hypotheses; a merging stage which suppresses redundant hypotheses; and a selection stage which chooses the best set of hypotheses that describe the data.

3.1. Exploration stage

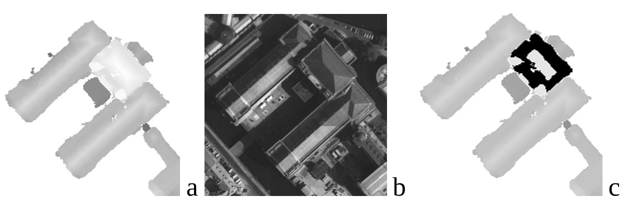


Figure 1. (*a*) is the initial raw DEM and (*b*) the ortho-image of the building. The black area in the other image is an example of extracted hypotheses by the exploration stage.

The purpose of this stage is to produce a list of possible planar surfaces of the building (hypotheses). All the different parts of the final model of the building must be found in this stage. The exploration stage is based on the RANSAC procedure (RANdom SAmple Consensus). We adapted this procedure to search the model hypotheses which describe the different parts of the data (see figure 1).

The exploratory procedure is iterative and each step can be described as follows: (1) randomly select a minimal set of points to initialize a model hypothesis, (2) grow this subset with consistent data and reject invalid points, and (3) test the validity of the model hypothesis (if the support set exceeds a threshold). With a simple planar patch model, the minimal set of points needed to construct a plane is defined by three non-colinear points. We set the number of hypotheses to search at 50.

The minimal set of points is selected as follows: the first point is randomly chosen from the DEM. The two others are chosen from a small window centered at the first point. The growing technique uses a search process that looks for candidates near the plane determined by fitting the current hypothesis. The candidates should be neighbors in 2-D DEM coordinates. We also use a recency map to conduct the exploration of the scene. When we find a valid model hypothesis, we store it in the recency map for a finite number of iterations (20 iterations). The values in the map are decreased after each initial random sampling even if there is no valid hypothesis. The random selection of the initial set of points cannot take points which are in the recency map.

3.2. Merging stage

We use a merging stage that reduces the redundancy in the list of hypotheses before the selection stage. We merge two hypotheses if they have a significant overlapping surface, or if there is a high probability that they correspond to the same surface. We estimate the overlapping surface by using the number of common points of the two planar patches. Surfaces with 80% overlap are merged. The second condition for merging is based on the statistical F-TEST. This statistical test compares the variances of two samples of data by maximizing the rejection of the equivalent case. We compute the probability that the combined patches is better than the individual. If these probability is greater than 0.9, we merge the two hypotheses.

3.3. Selection stage

The purpose of the selection stage is to decide which hypotheses must be kept. We want to remove the randomness of the exploration stage and select the minimum and the best set of hypotheses. The RANSAC procedure has it own selection stage to select the best model of the list. We however propose a different selection process to find the best set of models (i.e. planar patches) that describes the building. This selection stage is performed by casting the selection problem as an optimization problem. We used a solution based on the MDL principle (Minimum Description Length).

This stage decides whether to keep or to reject model hypothesis: this is a Boolean optimization problem. The number M of hypotheses in the list L is the size of the problem. Let the vector $m^T = [m_1, m_2, \ldots, m_M]$ be a set of models. m_i is a Boolean variable which expresses the presence ($m_i = 1$) or not ($m_i = 0$) of the model M_i within the solution m^T. The description length F value for the subset m is defined as follows:

$$\begin{aligned} F(m) &= \sum{}_{M_i \in \{L\}} m_i \cdot Q(M_i) \\ &+ \sum{}_{M_i \in \{L\}, M_j \in \{L - M_i\}} m_i \cdot m_j \cdot I(M_i, M_j) \end{aligned} \quad (1)$$

F is able to take into account the quality of a model and the pairwise interaction between the models. The function $Q(M_i)$ expresses the benefit value for a particular model M_i of the list, and $I(M_i, M_j)$ expresses the cost value of the interaction between the models M_i and M_j. F must be maximized to find the best subset of models. We take:

$$Q(M_i) = K_1 \cdot n_i - (1 - K_1) \cdot \Sigma_i \quad (2)$$

$$I(M_i, M_j) = \frac{(-K_1 \cdot |D_i \cap D_j| + (1 - K_1) \cdot \Sigma_{ij})}{2} \quad (3)$$

$\Sigma_{ij} = \max(\sum_{x \in R_i \cap R_j} d^2(x, M_i), \sum_{x \in R_i \cap R_j} d^2(x, M_j))$, $K_1 \in [0, 1]$ is a weight which allows us to adjust the preference of one of the two terms, n_i is the size of the support D_i of the model M_i and Σ_i is the sum of residuals. d is the Euclidean distance between a point x and a model M_i. $Q(M_i)$ favors the models from the list which have a large support and a small error measure. $I(M_i, M_j)$ limits the overlaps between the models of the subset that we are evaluating.

To solve this Boolean optimization problem, we need a discrete optimization procedure: we chose the Tabu search procedure described in [6]. Tabu search is a general heuristic procedure for global optimization which can be viewed as an extension of a steepest ascent method.

3.4. The segmentation system

We developed two modes of exploration for our experiments. In the first mode we can use very high resolution DEMs. We do not constrain the hypotheses (i.e. the planar patches), so that we can find all kind of roofs. With lower resolution DEMs, such as the one we use (at 50cm/pixel), the roofs are too coarse and the results of reconstruction may not be reliable. In these cases, we use a second mode of exploration, where hypotheses are constrained to correspond to horizontal planar patches. We tested the segmentation procedure with multiple estimators (traditional and robust). From the experiments, we adopted different methods for each of the two modes. With the horizontal constraint, we use the LMS estimator. In the unconstrained mode we use the LS estimator to keep the computational time low.

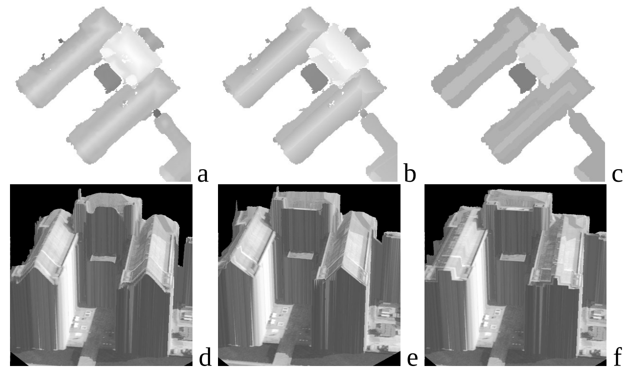


Figure 2. The pair of images (*a* and *d*) corresponds to the initial DEM and an arbitrary view of the building using this DEM and the ortho-photo, (*b* and *e*) corresponds to the synthetic DEM built using a segmentation in planar patches (22 patches selected) and (*c* and *f*) using the horizontal planar patches segmentation (11 patches selected).

Results are presented in figure 2. We chose a complex building to compare the results of the two modes of segmentation. Using a segmentation into planar patches of any orientation (22 patches) allows us to obtain a better visual 3D reconstruction of the building than using horizontal planar patches (11 patches), but it is not always reliable. The high level of outliers in the DEM disturbs the segmentation process. The presence of outliers requires an increase in the number of hypotheses which requires a greater computational time for each stage. The selection process becomes a more difficult task, which decreases the reproductibility and the quality of the results. For this reason, we prefer to use the horizontal planar patches segmentation, which gives robust result of reconstruction for buildings of all shapes.

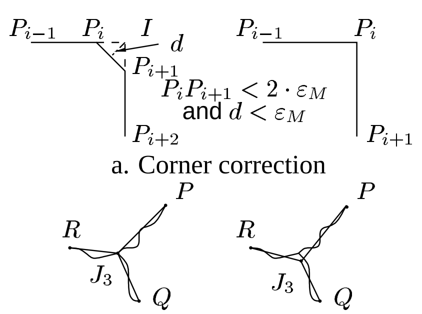
4. Polygonal model of the building

Once we have extracted each planar surface patch from the roof of the building, we want to obtain the polyhedral model of the building. Because we adopted a $2\frac{1}{2}$-D strategy to simplify the implementation and to give consistency to the final 3D model of the building, the 3D polyhedral model corresponds to a 2D polygonal model with an elevation value associated with each vertex. We propose a two-stage process. The first stage is the polygonalization of the contours of the selected hypotheses, and the second stage is an iterative refining procedure, which constrains some angles of the polygonal model to be right or straight.

4.1. Polygonal approximation of the building

Pre-processing We construct a synthetic local DEM from our list of models where each pixel is assigned to only one model. This synthetic DEM allows us to guarantee a $2\frac{1}{2}$-D consistency of the future polygonal model. If a pixel of the local DEM belongs to multiple models, the pixel is assigned to the model with the lowest elevation. If a pixel of the DEM does not belong to any model, we take the elevation value of the pixel from the raw DEM and assign this point to the model with the closest Z value in the neighborhood. Next, we apply a filtering procedure with two stages. First, we suppress the small regions, those with less than 50 pixels ($12.5m^2$). Second, we apply morphological filters to smooth the boundaries (*open/close* then *close/open*).

The synthetic local DEM that we obtained can be viewed as a segmented image. We propose a methodology for extracting the polygonal model from this segmented DEM. We begin by extracting two features from this image: the junctions and the chains. Chains are lists of successive points along the boundaries of the different regions. Junctions are the ends of the chains and can have different types: a *simple junction* is the intersection of the border of the DEM and a chain, a *double junction* closes a chain, and *complex junction* is at the the points where multiple regions meet. We present the framework in two distinct processes. The first process computes a polygonal approximation for each chain, with the junctions remaining fixed. The second process analyzes the different configurations of the junctions and adjusts their positions if necessary.



b. Processing of triple junction

Figure 3. Corner correction and junction processing. Left is before and right is after.

Polygonal approximation of individual chains Our algorithm for polygonal approximation of individual chains is based on the split and merge algorithm [4]. The original algorithm uses successive split and merge stages *while* the polygonal chain changes. Then, a Least Squares approximation stage estimates the parameters of each segment and updates the positions of the vertices.

We have enhanced the original algorithm with three main features: (1) We add in the *while* loop, with the split and merge stages, a new stage for corner correction. This correction handles the case where the corner is "rounded" and is described by two points instead of one (fig. 3a). (2) The fitting stage of the segments and intersection points is inside the *while* loop because this stage may still require further split and merge operations. (3) We use a Least Median of Squares (LMS) estimator to obtain a more robust and representative solution of segments. Note that some stages generate vertices that were not present in the original chain. To select the corresponding points in the original chain, we look for the nearest points in the original chain. These points are used to delimit the lists of points of the chain used for segment fitting.

Junction processing In the polygonal approximation process, the ends of the chains (the junctions) are fixed to avoid a disconnection in the polygonal model of the building. In this process, we adjust the positions of the junctions to obtain a more representative polygonal model. We process all the junctions at the same time. For each type of junction, we use a process based on Least Median of Squares. We randomly sample two points in the different chains, estimate the position of the junction point and compute residuals for all random-sets. Then, we select the solution which minimizes the median of residuals (fig. 3b).

4.2. Refining the model with angle constraints

We have extracted a polygonal model of the building using a segmented DEM. In this extraction, we have not assumed any *a priori* knowledge of the shape of the building. We obtain polygons with arbitrary angles. In man-made environments, however, straight and right angles are often present. We present next a process which tries to impose angle constraints on the global polygonal model of the building, still allowing for non-right or non-straight angles, using a method based on M-estimator.

The initial polygonal model Ω_i of the building consists of segments which are linked by junctions or vertices of the polygonal chains of the building model. Since we want to preserve the global consistency of the model, the strategy is applied to the global model. We approach the problem of orthogonalization by the optimization of an objective function O. The best solution Ω_b corresponds to the minimum of the objective function:

$$\Omega_b = \min O(\tilde{\Omega}) = \min \left(A(\tilde{\Omega}) + I(\tilde{\Omega}) \right) \qquad (4)$$

This objective function comprises two components: a component A which constrains angles to be 90° or 180° and a component I which relates the result to the initial data. We associate one angle for each point of the polygonal chains, two angles for the triple junction, and so on. The simple junctions are fixed because they correspond to borders of the images. Let Γ be the set of all the angle variables of the polygonal model, we have:

$$A(\tilde{\Omega}) = \sum_{\alpha \in \Gamma_{\tilde{\Omega}}} W_1 \cdot \sin(2\alpha) \qquad (5)$$

The component A allows us to force the polygonal model to have prefered angles ($0°$, $90°$, $180°$, and $270°$). The orthog-

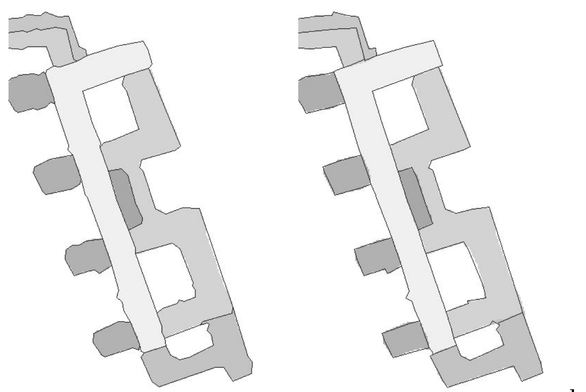


Figure 4. (*a*) polygonal approximation and (*b*) refinement.

onalization process only uses the polygonal model as input data. We need to use a component which relates result to the initial data and avoids large distortions on the polygonal model. Let Λ be the set of points of the polygonal model (junctions and vertices of the polygonal chains), we have:

$$I(\tilde{\Omega}) = \sum_{\beta \in \Lambda_{\tilde{\Omega}}} W_2 \cdot |\beta_i - \beta| \quad \text{with} \quad \beta_i \in \Lambda_{\Omega_i}$$

β is a point of the current polygonal model and β_i is the same point of the initial model. W_1 and W_2 are two weights which control the influence of the two components of the objective function. We choose $W_1 = 1/\sin(2 \times 10°)$ and $W_2 = 1/\varepsilon_M$ (ε_M is the threshold used in the merge stage of the polygonal approximation process) to have the same cost for a distance of ε_M from the initial model and for an angular difference of $10°$. Because we have an initial model close to the solution, we use the M-estimator method for the optimization with the Tuckey function. After optimization, we apply a merge iterative process to eliminate some of the straight angles or zero angles from the polygonal chains. The whole process, however, does not ensure that the optimized polygons do not intersect, since each chain or junction is considered separately. Though this situation did not occured in our experiments, a final stage should check and correct the global model consistency. Results are presented in figure 4. The orthogonalization procedure corrects most of the angles of the building.

5. Results

The results of the modeling system are presented in figure 5. We applied the process on a 1km$\times$1km area of the city of Berlin. The initial DEM has a ground resolution of 50cm. The results presented in the previous figures [1-4] were obtained with an error tolerance threshold of 2 meters in the exploration stage. This low threshold allowed us to show that the segmentation process can recover all the parts of the buildings. For figure 5, we used a threshold of 4 meters in to extract only the main components of the roofs.

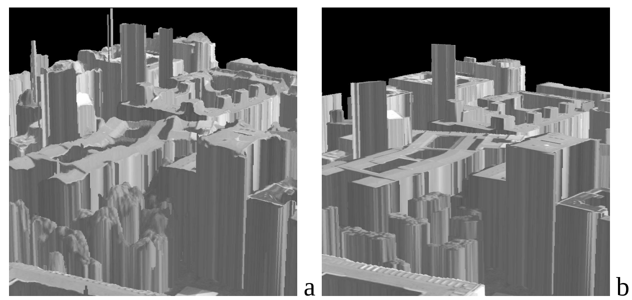


Figure 6. Comparison of the 3D views generated from the initial raw DEM (*a*) and from the output of the automatic building modeling process (*b*).

Figure 5*a* shows the results of the polygonalization stage. The model preserves the main structures of the buildings in the DEM. Figure 5*b* shows the final orthogonalized model. We recover most of the straight and right angles of the polygonal models. Figure 6 shows 3D views from the initial raw DEM and from the output of the automatic modeling process. Note that the reconstruction is a visually better representation of the scene. Using robust estimation techniques at the different stages of our global strategy allowed us to recover a consistent and representative model of each building. The computing times on a Sun ultra sparc 10 are about 25 minutes for the complete segmentation of the buildings, 4 minutes to extract the polygonal models of the buildings and 20 minutes for the polygonalization.

In one application, the digital models provide inputs to planning tools for wireless networks. These tools simulate the coverage of a cell in the city to help reduce the number of survey measurement needed. To validate the results of our automatic system, we compared in another report to appear the simulated digital models, obtained from different

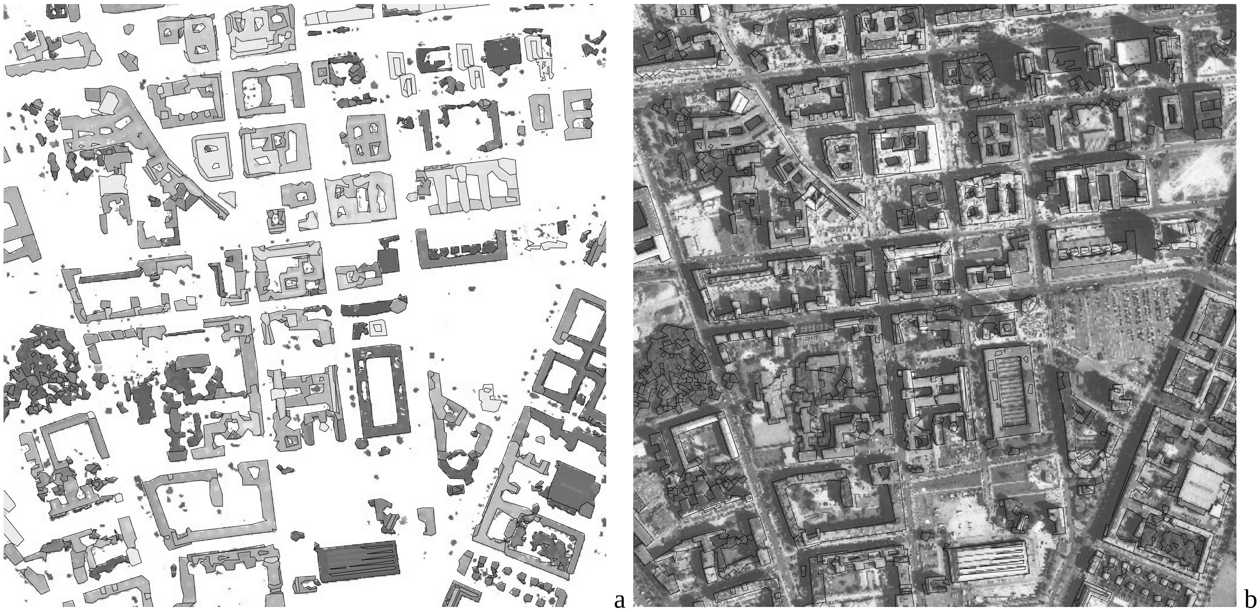


Figure 5. Berlin results of automatic building extraction: (*a*) the polygonal approximation result superimposed on the DEM composed by all the extracted objects above the ground (small components are then discarded). (*b*) the final orthogonalized model superimposed on the ortho-image of the scene. Note that the model describes well the main structures of the buildings.

methods (the initial raw DEM, the automatic model build by our method and a manual process), with reference model from a survey. Results show that the quality of the results of simulation with the automatic DEM is similar to those obtained with the manual DEM.

6. Conclusion

We presented a system for modeling buildings from a single Digital Elevation Model (DEM). This system uses various robust estimation methods to extract the main representative components of the building despite a large amount of noise in the DEM. We construct the polygonal model of the building in two stages. The first stage segments the DEM into planar surface patches for describing the building. Then, the polygonalization stage generates the final polygonal model of the building by using weak constraints. This system is fully automatic and does not use any a priori information about the shape of the buildings.

We presented results from a scene with multiple buildings in a 1km$\times$1km area of Berlin. The polygonal model is shown to correctly represent the buildings in the scene. The performance of the system depends on the quality of the initial DEM. In another report, the result was also validated against a mobile network planning application, and using the result of our methods showed large improvements in quality over using the initial raw DEM.

References

[1] B. Ameri and D. Fritsch. 3-d reconstruction of polyhedral-like building models. In *ISPRS Conference : Automatic Extraction of GIS Objects from Digital Imagery*, pages 15–20, Munich, Germany, 1999.

[2] M. Fradkin, M. Roux, and H. Maître. Building detection from multiple aerial images. In *ISPRS Conference : Automatic Extraction of GIS Objects from Digital Imagery*, Munich, Germany, April 1999.

[3] H. Mayer. Automatic object extraction from aerial imagery–a survey focusing on buildings. *Computer Vision and Image Understanding*, 74(2):138–149, May 1999.

[4] T. Pavlidis and S. L. Horowitz. Segmentation of plane curves. *IEEE Transactions on Computers*, 23(C):860–870, 1974.

[5] L. Gabet, G. Giraudon, and L. Renouard. Automatic generation of high resolution urban zone digital elevation model. *ISPRS: Journal of Photogrammetry and Remote Sensing*, 52(1):33–47, February 1997.

[6] M. Stricker and A. Leonardis. Exsel++ a general framework to extract parametric models. Technical Report BIWI-TR-159, ETH Zürich, February 1995.

[7] Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing Journal*, 15(1):59–76, 1997.