Using robust methods to model buildings with a DEM

Christophe Vestri, Frédéric Devernay, Marion Mesnage

USING ROBUST METHODS TO MODEL BUILDINGS WITH A DEM

Christophe Vestri, Frédéric Devernay, Marion Mesnage
ISTAR
2600, route des Crêtes, BP 282
06905 Sophia-Antipolis CEDEX
Ph.: +33-(0)4-97-23-23-39
E-mail: vestri@istar.fr, devernay@unice.fr, bignone@istar.fr

KEY WORDS: Modeling, Building, DEM, Robust Estimation, Constrained Polygonalisation

ABSTRACT

In this article we present a framework for modeling a building with a single Digital Elevation Model (DEM). The model is constructed in two stages. A first stage segments the DEM into planar surface patches to describe the building. Then, a polygonalisation stage generates the final polygonal model of the building by using constraints. We use robust estimation methods at different stages of our framework to develop an efficient and insensitive to noise system of modeling. The system that we propose is fully automatic and does not use any a priori information about the shape of the buildings. We present results with simple buildings and with a large area of a real city.

1 INTRODUCTION AND RELATED WORKS

Extracting building descriptions from an urban scene in 3D is an essential task for a variety of applications such as telecoms, urban planning, cartography, etc. Because of the difficulty of this task (complexity, number and diversity of 3D objects of the urban environment), this is usually done by human operators. The cost and time involved are high in manual reconstruction of buildings, therefore there is much active research on automatic 3D detection and reconstruction of the buildings. In this context, we present in this paper a system for the automatic modeling of buildings with a single Digital Elevation Model (DEM).

A variety of methods have been used for building reconstruction, namely building segmentation using a DEM, feature grouping in 2D or 3D, image segmentation or combined strategies. H. Mayer presented in (Mayer, 1999) a survey on automatic object extraction from aerial imagery and emphasized the complexity of this difficult problem. We can cover the knowledge of the field into model and strategy approaches. The Model approaches integrated into the model some knowledge about the 3D real world, geometric or topologic regularity of the scene or the spatial context. In these approaches, we recover the building model from images. The Strategy approaches are those that use a strategy to construct the model. This strategy can be either grouping, matching of primitives in multiple images, robust approximation of hypotheses extracted from a DSM. This list (Bignone et al., 1996), (Nevatia and Huertas, 1998), (Weidner and Förstner, 1995), (Stilla et al., 1996), (Collins et al., 1996) and (Moons et al., 98) is neither complete or exhaustive but represents the main publications in this domain .

2 MOTIVATION AND OVERALL STRATEGY

Our global strategy for modeling a building or a set of buildings is composed of three main stages. In the first stage, we construct a dense and reliable DEM using a correlation-based stereo method. The second stage is the segmentation of this DEM into locally planar surfaces. The objective of this second stage is to describe the scene using surface patches which correspond to the various facets of the buildings. The third and last stage is the vectorization of the boundaries of each surface patch to obtain the final model of the buildings. The first stage is presented in (Vestri and Devernay, 2000). The framework that we propose in this article corresponds to the two last stages of our global strategy, the DEM being the input of our framework. We propose to use several robust methods to solve the complex problems of this framework. We begin by a quick review of the various robust methods for parameter estimation.

3 ROBUST METHODS FOR PARAMETER ESTIMATION

Classic methods for parameter estimation suppose that the data are corrupted by a small-scale noise. If there are gross errors (outliers) in the data, these methods can produce nonsense results. Robust methods are designed to estimate correctly the model of the dominant population even if there is an important amount of outliers. These methods have been proposed and developed both in statistics and computer vision domains. We propose in this section to present the robust methods that we used. We orient the readers to the review of (Zhang, 1997) for more precisions.

The two main robust methods used in computer vision are M-estimator and Least-Median of Squares (LMS). The M-estimator is a generalization of Least Squares (LS) estimator. LS estimator tries to minimize the sum of the squared residuals which is unstable in the presence of outliers. The M-estimator reduces the effects of outliers by replacing the squared residuals by another function of the residuals. This function reduces the importance of data that have a high residual. Like LS estimator, M-estimator requires a good initial estimate to converge to the optimal solution. The principle of the Least-Median of Squares estimator is to select the best model from a set of estimated models that are randomly sampled. A crucial parameter of this method is the number of sets to estimate. The method chooses the model which has the minimum median squared residual. This method can recover the model with the presence of 50% of outliers. The computational time can increase dramatically with an important number of data.

4 DETECTION OF BUILDINGS

All the views of the scene are supposed to be calibrated. The initial DEM is made by robust fusion of several DEMs, each of them corresponding to the matching of two adjacent views. To keep the computational time low, the whole process is applied only to the neighborhood of a building, then we merge all results to obtain the final model of the scene. We automatically detect and extract, from the raw DEM, each building or group of adjacent buildings. For this detection, we substract the Digital Terrain Model (DTM which corresponds to the ground) of the scene from the raw DEM to obtain a height map. Then, we extract a local DEM for each blob in our height map by using a threshold. We keep only the buildings which are higher enough and have a sufficient size. We apply the segmentation and vectorization processes to each building (the local DEM) independently.

5 SEGMENTATION OF THE DEM

The first objective is to extract a simple and representative description of each building of the scene without any previous knowledge of their shape. This is the segmentation problem that we propose to solve. By using a DEM as initial data, this problem can be viewed as modeling a cloud of 3D noisy data. Many techniques to solve this problem use the model selection and the robust estimation methods. Recent reviews of these two domains are the paper of P. Torr (Torr, 1999) for the model selection and the paper of C. Stewart (Stewart, 1999) for robust estimation.

Our approach is based on the *ExSel++* framework presented in (Stricker and Leonardis, 1995). The authors define in this article a general and robust framework to extract parametric models from dense or sparse data. One specificity of their framework is the ability to use and select multiple models to describe the data. The DEM is a $2D^{\frac{1}{2}}$ map. Data from this map mainly correspond building roofs and ground. We choose the planar surface patch model to describe data in our segmentation process. We are able to describe all the buildings of the scene with this simple model, except some which have second order surfaces in their structure (dome, cylindric shape, ...). The segmentation process is composed of three main stages that we will present independently: a stage of exploration of the data which generates a list of hypotheses of models, a stage of merging which suppresses redundancy of the hypotheses and a stage of selection which chooses the best set of hypotheses to describe the data.

5.1 Exploration stage

The purpose of this stage is to produce a list of hypotheses which can later be used by the selection stage to describe the data. 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) which was proposed in (Fischler and Bolles, 1981). Like Least-Median of Squares (LMS, see (Rousseeuw and Leroy, 1987)), this method computes a model by solving a system of equations defined for a randomly chosen subset of points. All data are then classified relatively to this model and points which are in the error tolerance are called consensus set of the model. If the support of this consensus set exceeds a threshold, the model is validated and then is recomputed. We adapt this procedure to search and compute the hypotheses of model which will describe the different parts of the data.

The exploratory procedure is iterative and each step can be described as follows:

- randomly select a minimal set of points for the model,
- grow this subset with consistent data and reject invalid points and
- test the validity of the model hypothesis.

With a simple planar patch model, the minimal set of point to construct a plane is defined by three points which are not lined up.

We implement two additional features to conduct the exploration procedure and to improve the consistency of the results. First, we choose a deterministic growing from the initial set of data that we randomly select in the scene. Actually, only the first point of the minimal subset is randomly chosen in the scene. The two others are chosen in a restrictive window centered on the first point. We validate the three initial points with two verifications: we verify that there are not lined up and we verify that they do not correspond to a near-vertical plane. We take advantage of the $2D^{\frac{1}{2}}$ map to conduct the sampling and the growing.

The second feature is the usage of a recency map to conduct the exploration of the scene. When we have found a valid hypothesis of model, we store this hypothesis in this recency map for a finite number of the procedure 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.

We developed two modes of exploration for our experiments. In the first mode, we constrained the hypotheses of model to be horizontal features. In this case, we simply compute the median value of the altitudes to estimate the parameters of the planar surface patch. In the second mode, the planar surface patch is not constrained. We use a simple Least Squares estimator to compute the parameters of the planar surface patch. We can find all kind of roofs. To ensure a quick convergence of the estimation, the first estimated plane is constrained to be horizontal.

We use two stopping conditions for our exploratory procedure. First, the procedure is stopped when we have found enough hypotheses. Second, the procedure is also stopped when we cannot find another hypothesis from the data (the points which are not in the recency table). The first stopping criterion depending on the complexity of the scene. We evaluate the number of hypotheses to search at 50. The second stopping criterion is empirically chosen at 50 failed samples. The error tolerance threshold depends on the resolution of the DEM (see section 7.1).

5.2 Merging stage

After the exploration stage, we propose a merging stage in the segmentation process. This stage allows to limit the redundancy in the list of hypotheses and decrease the computational time of the selection stage. The remaining hypotheses after this stage are generally more consistent. The principle of this stage is to merge two hypotheses if they have an important overlapping surface or if there is a high probability that they correspond to the same surface.

The implementation of the first condition test is simple. We estimate the overlapping surface by using the number of common points of the two planes. We express this surface as a percentage in the plane which has the smallest surface (eg. the smallest number of points). The use of the percentage value avoids the confusion between two planes

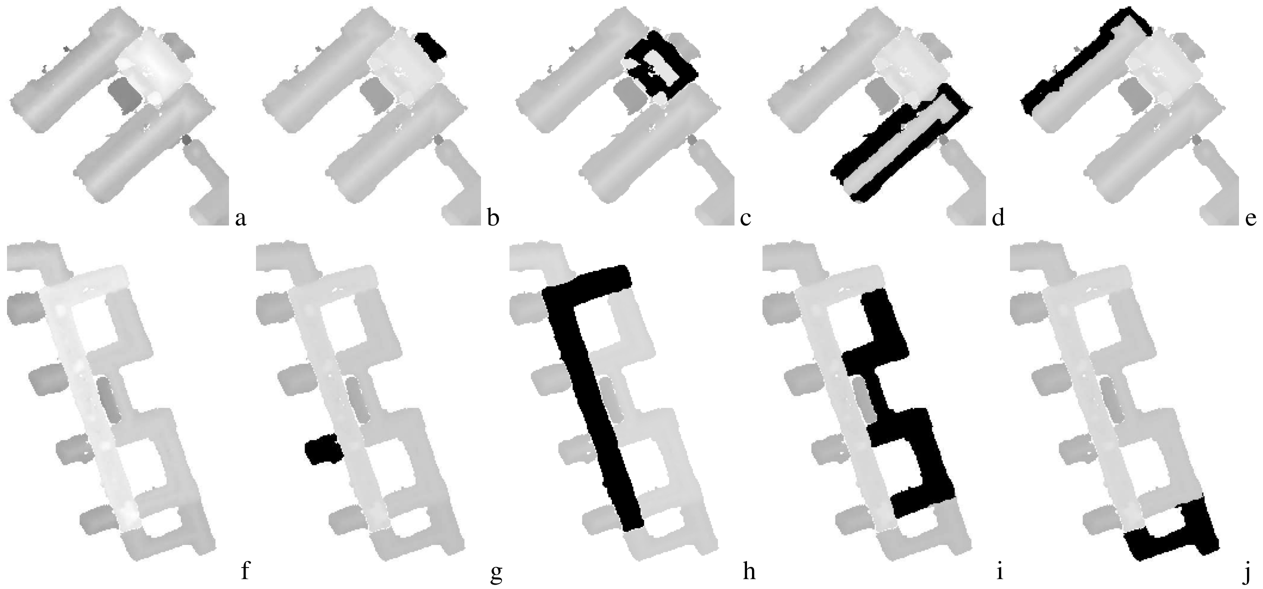


Figure 1: *a* and *f* are two initial local DEMs of buildings. The black areas in the other images are examples of hypotheses which are extracted by the exploration stage.

which correspond to the same surface and an intersection of two different planes. We use a decision threshold of 80% to determine if the planes must be merged.

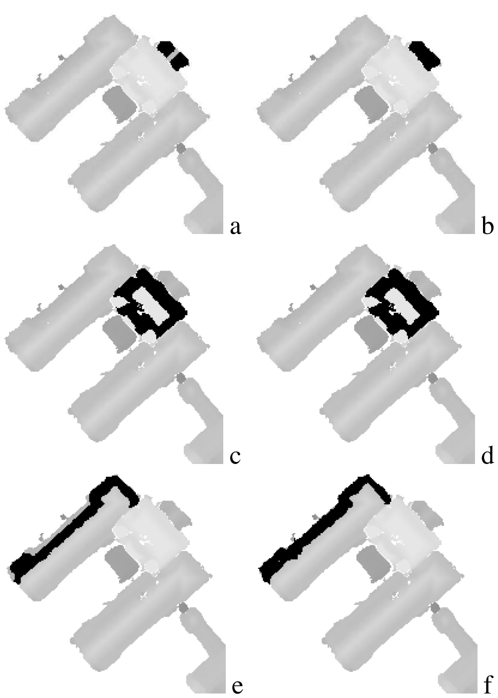


Figure 2: Merging of the hypotheses: Each of the different pairs of images (a b), (c d) and (e f) presents two hypotheses (black regions on the building image) that correspond to the same plane. These hypotheses are merged.

The second condition of merging is based on a statistic test: the F-TEST. This test is only used in the unconstrained mode of the exploratory searching. We use it to decide if the resulting plane of the combination of two hypotheses of planes is better than each of them in the sense of the test. The estimator of the parameters of the plane is unbiased. We use the variance of the error of the data with their model to quantify the quality of each hypothesis.

The F-Test allows to compare the variances of two samples of data by maximizing the rejection of the equivalent case. The advantage of this test is to compare the variance of two samples of data which have different sizes. We use the F-test to compare individually the quality of the combined plane with the quality of each of the two plane. We compute the probabilities that the combined plane is better than each of the two other planes. If these probabilities are upper than 0.9, we merge the two planes.

5.3 Selection stage

The purpose of the selection stage is to decide which hypotheses must be kept to describe the data. We want to remove the randomness of the exploration stage and select the minimum and the best set of hypotheses. The selection stage is performed by changing the selection problem in an optimization problem. We adopt the solution proposed by *A. Leonardis* in (Leonardis, 1994) which implements the MDL principle (Minimum Description Length).

The description length functions The MDL principle is based on the notion of performing inductive inference by minimal coding. In the selection problem, we want to select the minimal number of models which are necessary to describe the data. We also want to preserve a minimal measure of error between the data and the selected models. We use an objective function F, computed for a subset of model hypotheses of the list of hypothesis L. This function is a combination of two components: the first component $Q(M_i)$ which expresses the benefit value for a particular model M_i of the list, and a second component $I(M_i, M_j)$

which expresses the cost value of the interaction between the models M_i and M_j.

The component $Q(M_i)$ expresses the quality of an hypothesis of the list. This quality measure is composed by two terms: a term of benefice and a term of cost. The benefice term expresses the importance of the hypothesis. It depends on the size n_i of the support D_i of the model M_i (eg. the number of data use to generate this hypothesis: $n_i = |D_i|$). The cost term expresses the quality of description of the hypothesis. It depends on the measure of error Σ_i of the data which support the model M_i (Σ_i is the sum of residuals). We compute the quality component $Q(M_i)$ as follows:

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

with $K_1 \in [0, 1]$. We obtain a simpler component than *A. Leonardis* because we only use one kind of model. K_1 is a weight which allows us to adjust the preference of one of the two terms. The quality component allows to select the models from the list which have a big support and a little measure of error.

Because we have an overlap between the different models, we need to take into account this interaction in the optimization to limit this phenomenon. Like *A. Leonardis*, we only consider pairwise overlaps of the models. We use an interaction component $I(M_i, M_j)$ between two models which have the same form as the quality component. These component is evaluated on the overlapping part of the two models. But because we want a minimal overlap between the models, the terms of the interaction component are opposed to the terms of the quality component. We compute the Interaction component $I(M_i, M_j)$ as follows:

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

with:

$$\Sigma_{ij} = \max\left(\sum_{x \in R_i \cap R_j} d^2(x, M_i), \sum_{x \in R_i \cap R_j} d^2(x, M_j) \right)$$

d is the Euclidean distance between a point x and a model M_i. The interaction component allows to limit the overlaps between the models of the subset that we are evaluating.

The boolean optimization problem Each hypothesis of model must be selected or not in this stage, this is a boolean optimization problem. The number M of hypothesis in the list L is the size of the problem. Let vector $m^T = [m_1, m_2, \ldots, m_M]$ denote 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 in the solution m^T. The description length F value for the subset $\tilde{m}$ is defined as follows:

$$\begin{aligned} F(\tilde{m}) &= \sum_{M_i \in \{L\}} m_i \cdot Q(M_i) &(3)\\ &+ \sum_{M_i \in \{L\}, M_j \in \{M - M_i\}} m_i \cdot m_j \cdot I(M_i, M_j) \end{aligned}$$

F must be maximized to find the best subset of models. We solve this as a quadratic boolean problem. The objective function to be maximized is the following :

$$F(m) = m^T R m \qquad (4)$$

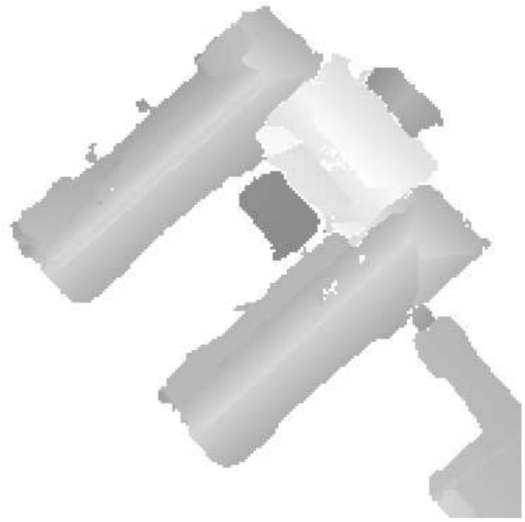


Figure 3: Results of the selection stage: The segmentation procedure uses 22 planes to describe the building (in the mode that does not use the horizontal constraint).

This function allows to take into account the quality of a model and the interaction between all of them (by pair). The diagonal terms of the matrix R express the cost-benefit value for a particular model M_i, we take the quality component $Q(M_i)$. The off-diagonal (i, j) terms handle the interaction between the overlapping models M_i and M_j, we take the interaction component $I(M_i, M_j)$. The matrix R is symmetric. Because off-diagonal terms depend on the overlap of the models, the matrix can be sparse or banded.

Tabu search To solve this boolean optimization problem, we need a procedure of discrete optimization. We use the Tabu search procedure to solve the system. Tabu search is a general heuristic procedure for global optimization which can be viewed as an extension of a steepest ascent method. We do not describe this algorithm in this article. The idea of Tabu search is to begin the steepest ascent from multiple initial conditions. For each iteration, we evaluate the objective function for the current subset of selected models and for the neighbor subsets and then we choose the best move. We save all the local maxima found by the procedure in a table. The global solution corresponds to the subset of models with the maximum value of all the local maxima. We implemented this procedure with the components proposed by (Stricker and Leonardis, 1995)

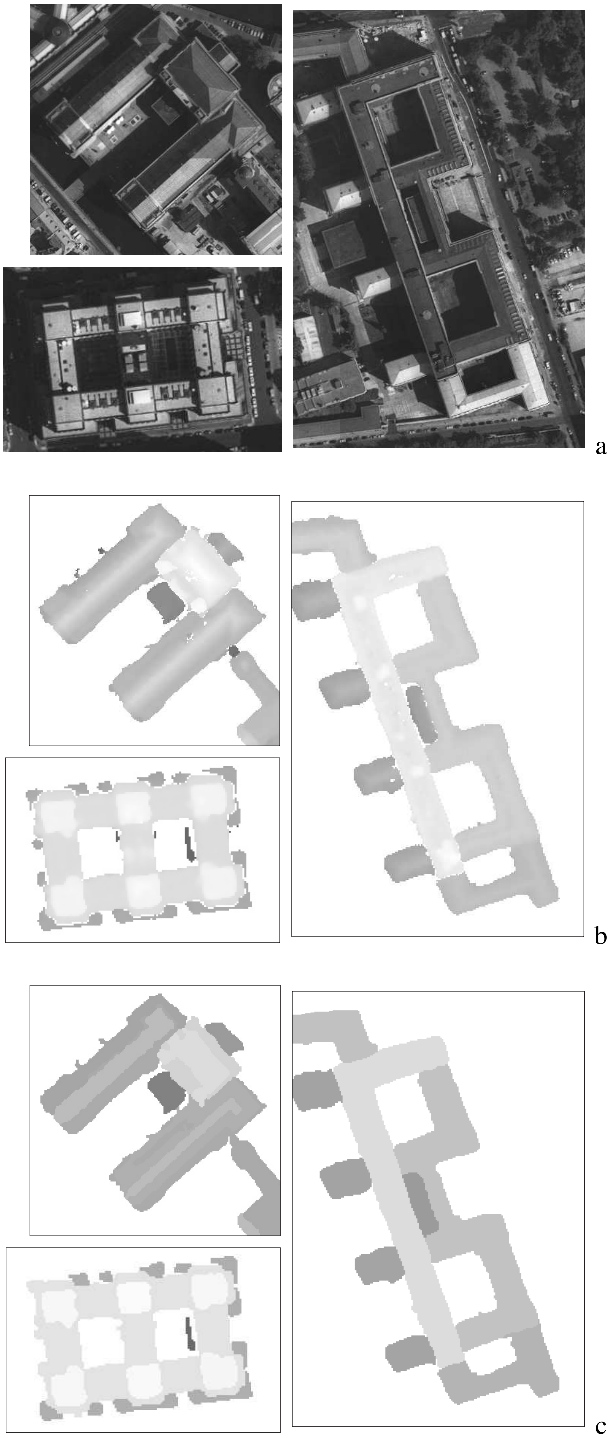


Figure 4: The images of *a* present three ortho-images of buildings. The images of *b* are the corresponding local DEMs. The synthetic local DEMs of the building that we construct after the segmentation process are presented in *c*. We only use horizontal planes to describe the buildings.

6 CONSTRUCTION OF THE POLYGONAL BUILDING MODEL

Once we have extracted each planar surface patch from the roof of the building, we want to obtain the polygonal model of the building. We adopt a $2D^{\frac{1}{2}}$ strategy to simplify the implementation and to give consistency to the final 3D model of the building. We propose a strategy in two main stages. The first stage is the polygonalisation of the contours of the selected hypotheses. The second stage is an iterative procedure of refining. This procedure of refining constrains angles of the polygonal model to be orthogonal or straight.

6.1 Polygonal approximation of the building

The list of planar surface patches, generated by the segmentation process, describes the different parts of the surface of the roof. There are two problems we need to solve before we apply our polygonal approximation: first, some overlapping regions remains between the patches and second, some holes appear between the models and in the middle of the models. Most of these holes come from the initial DEM, where the matching process failed. We begin the polygonal approximation stage with a pre-treatment stage to correct these problems. Then, we present a framework for extracting a polygonal model with a segmented DEM.

6.1.1 Pre-treatments We propose to construct a synthetic local DEM from our list of models where each pixel is affected to only one model. This synthetic DEM allows to guarantee a $2D^{\frac{1}{2}}$ consistency of the future polygonal model. If a pixel belongs to multiple models, the pixel is affected to the model with the lowest altitude. We choose this mode of affectation because roofs are generally out of their real limits. If a pixel does not belong to any model, we use another complete DEM. This DEM corresponds to the interpolated map of the local DEM. We take the altitude value of the pixel into the complete DEM. Then, we affect this pixel to the model which is the nearest (in Z component) from this altitude.

After the construction of this synthetic and complete DEM, we apply a filtering procedure with two stages. The first stage suppresses the small and residual regions which have a surface inferior than $12m^2$. The second stage is a morphologic filtering. We use this filtering for smoothing the boundaries between the different models and mainly near the junctions. We use the *open/close* then *close/open* morphologic filters.

The synthetic local DEM can be viewed as a segmented image. We propose a framework 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 which delimit the different regions. Junctions are the limits 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 a *triple or higher degree junction* corresponds to the limit of multiple regions. We present the framework in two distinct processes. The first process does a polygonal approximation of each chain, junctions are fixed. The second process treats the different configurations of the junctions and simplifies the model if this is necessary.

6.1.2 Polygonal approximation of the chains

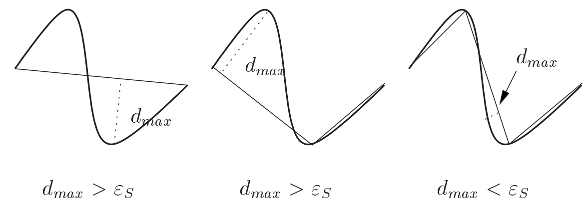
We propose an algorithm of polygonal approximation of chains based on the work by (Pavlidis and Horowitz, 1974). This choice was conducted by a comparative study of polygonal approximation methods presented in (Filbois, 1995). The original algorithm uses successive split and merge stages while the polygonal chain changes. Then, a Least Squares approximation stage estimates the parameters of each segments and the final stage compute the new positions of the points of the polygonal chain. We adopt the same merging and splitting tests in our algorithm. The chain is splitted if d_{max} the maximum distance of the points of the chain to the current polygonal chain is upper than a threshold ε_S (fig. 5a). Two successive segments of the polygonal chain are merged if the distance d between the middle point (which is at the intersection of them) and the straight line defined by the other points of the polygonal chain is greater than a threshold ε_M (fig. 5b).

We propose to improve the original algorithm of Pavlidis with the next features:

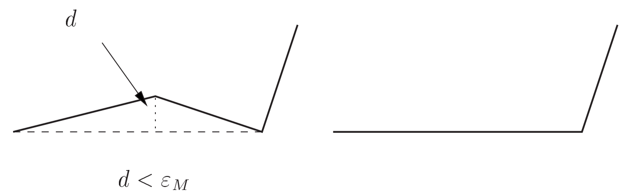
- We add another merge criteria based on the surface of the triangle that is defined by three successive points of the polygonal chain. It allows to suppress residual noise in the polygonal chain (fig. 5c).
- We add in the *while loop* with split and merge stages a new stage of corner correction. This correction treats the case where angles are too smooth and the chain is described by two points instead of one (fig. 5d).
- The fitting stage of the segments and intersection points is put in the *while loop* because this stage can still require split and merge stages.
- We use a Least Median of Squares (LMS) instead of Least Squares (LS) estimator to avoid initialization problems and to obtain a more robust and representative solution of segments.

We describe our algorithm in the figure 6.

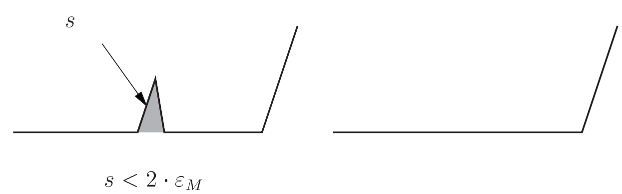
Because we add the correction of the corners, the estimation of the segments and the computing of intersection of segments stages in the loop, the points of the polygonal chain can be out of the chain. For selecting the representative points of the chain, we search for each point of the polygonal chain the nearest points of the chain. We use these points to delimit the lists of points that we use to fit a model of segment (fig. 5e).



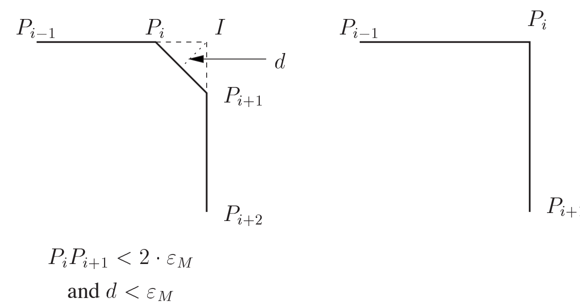
a. Split stage: while $d_{max} < \varepsilon_S$, the chain is splitted



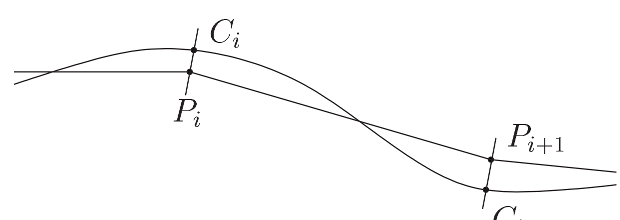
b. Merge with the distance criteria.



c. Merge with the surface criteria.



d. Correction of the corners: if the two tests are valid in the polygonal chain, we recover a corner of the model and then correct the polygonal chain.



e. Estimation of parameters of a segment with the chain: we use the list of points of the chain which are between C_i and C_{i+1} to estimate the parameters of the segment P_iP_{i+1} of the polygonal chain.

Figure 5: *a*, *b*, *c* and *d* present the different procedures and tests of the polygonal approximation process. *e* present the choice of the list of points of the chain that we use for estimating a segment model of the polygonal chain.

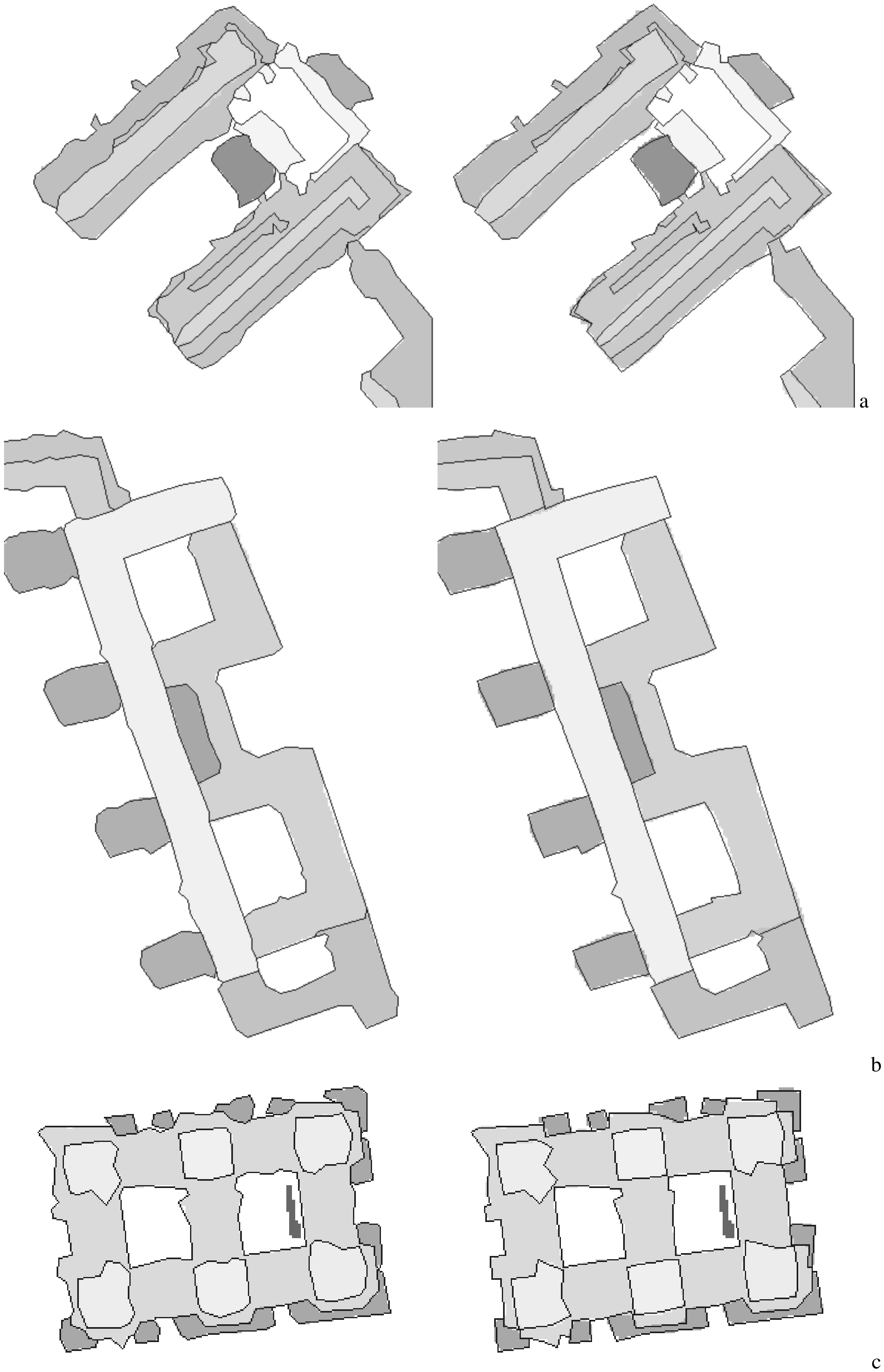


Figure 8: *a*, *b* and *c* present the vectorization of three buildings. The first images correspond to the result of the polygonal approximation of the chains. The second images are the results of the refining process.

---

while modification by split or merge phase

- split phase of the chain based on distance between chain and polygonal chain
- merge phase of the chain based on distance between chain and polygon chain and surface between three consecutive polygonal points
- correct the corners
- adjust segments of the polygonal chain with LMS estimation
- compute new intersections of segments to obtain the final polygonal chain

end while

Figure 6: Algorithm of polygonal approximation of the chains.

6.1.3 Treatment of the junctions In the polygonal approximation process, the limits of the chains (the junctions) are fixed to avoid a deconnection in the polygonal model of the building. In this process, we want to adjust the positions of the junctions to obtain a more representative polygonal model. We treat all the junctions at the same time. Each type of junction has an adapted process that we present in the figure 7.

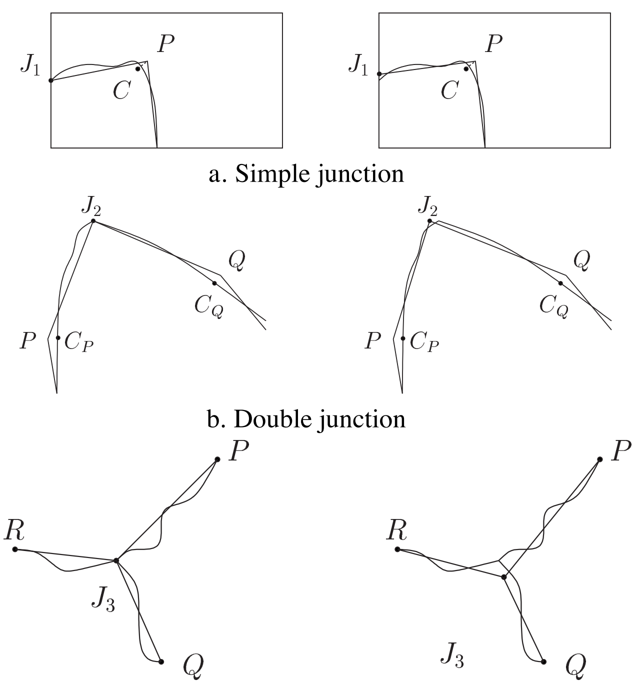


Figure 7: *a*, *b* and *c* present the treatments of the different junctions to correct their positions in the polygonal model.

A simple junction point J_1 is the intersection of the border of the DEM and a polygonal chain. First, we search the points of the chain which correspond to the current segment. Like in the polygonal approximation process, we compute the nearest point C of the chain from the other extremity of the segment P. We compute the new segment by fitting a segment model which passes through P to the points of the chain which are between J_1 and C. Then, we move the junction point to the intersection of the new segment and the border of the image.

A double junction point J_2 corresponds to a closed polygonal chain. We use the same strategy with two segments. We compute the two parts of the chain J_2C_P and J_2C_Q which respectively correspond to the two segments J_2P and J_2Q. We fit with a LMS estimator the two segments which pass through the initial extremities P and Q. Then, we move the junction at the position of their intersection.

For triple junction points, we compute the junction position which minimizes the sum of residuals of the three segments starting from the junction. We search the best position of the junction when other extremities do not move. Our LMS strategy consists in randomly sampling two points in two different chains. We 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. We do not treat the junctions with a highest degree.

6.2 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 supposed any *a priori* knowledge on the form of the building. We obtain polygons with any angles. In man-made environments, straight and orthogonal angles are often present. We propose now a process which tries to impose these constraints to the global polygonal model of the building.

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

$$\Omega_b = \min O(\tilde{\Omega}) = \min \left(A(\tilde{\Omega}) + I(\tilde{\Omega}) \right) \qquad (5)$$

We explain in the following the construction of this objective function which comprises two components : a component A which constrains angles to be 90° or 180° and a component I which attaches the result to the initial data.

The first component gives priority to 90° and 180° angles. We use the $sin^2(2\alpha)$ function to estimate the weight of an angle α of the polygonal model. This function is minimal for the angles we want to privilege (0° MODULO 90°). We use one variable angle for each point of the polygonal chains, two variable angles for the junction triple, etc. The simple junctions are fixed because they correspond to borders of the images. Let Γ be the ensemble of all the angle's variables of the polygonal model, we have:

$$A(\tilde{\Omega}) = \sum_{\alpha \in \Gamma_{\tilde{\Omega}}} W_1 \cdot sin^2(2\alpha) \tag{6}$$

The component A allows to force the polygonal model to have privileged angles.

The orthogonalization process only uses the polygonal model as input data. We need to use a component which attaches result to the initial data and avoids too important distortions on the polygonal model. Let Λ be the ensemble of the points of the polygonal model (junctions and vertices of the polygonal chains), we have:

$$I(\tilde{\Omega}) = \sum_{\beta \in \Lambda_{\tilde{\Omega}}} W_2 \cdot |\beta_i - \beta| \tag{7}$$
$$\text{with} \quad \beta_i \in \Lambda_{\Omega_i}$$

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 $W_1 = 1/\sin(10°)$ and $W_2 = 1/\varepsilon_M$ (ε_M is the threshold using 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 closed to the solution, we use the M-estimator method for the optimization with the Tuckey function. After the optimization, we use a merging stage to eliminate some bad configurations of the polygonal chains due to constraints.

7 RESULTS

7.1 Segmentation of the DEM

The results of the segmentation process are presented in the figure 9. We apply the process on an area of the city of Berlin. The initial DEM has a ground resolution of 0.5 meters. All the past results that we have presented in the figures [1-8] have been constructed by using an error tolerance threshold of 2 meters in the exploration procedure. This low threshold allowed us to show that the segmentation process can recover all the planar patches to describe the building. In the figure 9 and for all results that we present in the area of Berlin, we use a threshold of 4 meters for extracting only the main components of the roofs. We can see in the figure 9 that the segmentation process has extracted the different parts of the roofs of each building of the scene.

We test the segmentation procedure by combining multiple estimators (traditional and robust) with different estimation procedures. The estimators that we use are a Least Squares (LS) estimator, a Non Linear Least Squares (NLS) estimator, a robust M-estimator and the Least-Median of Squares (LMS) estimator. The different procedures that we use for estimations of the models are: (1) search all the points that verify the initial model then refine the model, (2) after each growing stage, refine the parameters of the model and (3) after each growing stage, refine the parameters of the model and reject the points which are not in the tolerance domain of the new estimate model. From the experiments, we adopt different methods for each of the two modes: segmentation of the building with horizontal planes or with planes which can have any orientation. With the horizontal constraint, we use the LMS estimator and the third procedure which allows to suppress the erroneous data along the procedure. In the not constrained mode we adopt the third procedure with LS estimator to keep the computational time low.

7.2 Polygonalisation of the contours

The figure 10 presents the results of the polygonal approximation stage. The results of the orthogonalization stage are presented in the figure 11. We can see in the figure that we recover most of the straight and the orthogonal angles of the polygonal models. We also preserve the main structures of the buildings of the DEM. 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 an Sun ultra sparc 1 are about 1 hour for the complete segmentation of the buildings, 8 minutes to extract the polygonal models of the buildings and 3 hours for the polygonalisation.

8 CONCLUSION

We have presented in this article a framework for modeling a building with a single Digital Elevation Model (DEM). This framework uses multiple robust estimation methods to extract the main representative components of the building despite an important amount of noise in the DEM. We construct the polygonal model of the building in two stages. The first stage segments the DEM in planar surface patches for describing the building. Then, the polygonalisation stage generates the final polygonal model of the building by using constraints. This framework is fully automatic and does not use any a priori information about the shape of the buildings. We only constraint angles to be straight or orthogonal if this is necessary.

We have presented results with multiple buildings and with an area of Berlin. The polygonal model that we obtained represents correctly the buildings of the scene. The weakness of our framework is the dependence to the quality of the initial DEM. We are working on the improvement of the DEM to have a complete, efficient and automatic system for modeling buildings. We are also studying a new procedure of polygonalisation because the computing time is too important. We wish to improve the efficiency and decrease the computing time by using a procedure which randomly modifies the different angles, computes the quality of the configuration and then selects the best configuration of the model.

Figure 9: Results of segmentation: The segmentation process recovers all the planar patches of the roofs of the buildings.

Figure 10: Result of the polygonal approximation stage (black lines).

Figure 11: Result of orthogonalization of the polygonal model (black lines). The algorithm recovers most of the straight and the orthogonal angles of the polygonal models.

REFERENCES

Bignone, F., Henricsson, O., Fua, P. and Stricker, M., 1996. Automatic extraction of generic house roofs from high resolution aerial imagery. In: B. Buxton (ed.), Proceedings of the 4th European Conference on Computer Vision, Cambridge, UK, pp. 85–96.

Collins, R., Jaynes, C., Cheng, Y.-Q., Wang, X., Stolle, F., Schultz, H., Hanson, A. and Riseman, E., 1996. The umass ascender system for 3d site model construction. In: RADIUS: Image Understanding for Imagery Intelligence, pp. 209–222.

Filbois, A., 1995. Contributions à la modélisation automatique d'objets polyédriques 3D : extraction de primitives 3D, facettes et segments. PhD thesis, Institut National Polytechnique de Lorraine.

Fischler, M. and Bolles, R., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, pp. 381–385.

Leonardis, A., 1994. Recover and select paradigm–a robust approach to estimation of parametric models. In: Performance versus Methodology in Computer Vision NSF/ARPA Workshop, Seattle, WA, pp. 78–87.

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

Moons, T., Frère, D., Vandekerckhove, J. and Gool, L. V., 98. Automatic modelling and 3d reconstruction of urban house roofs from high resolution aerial imagery. In: Proceedings of the 5th European Conference on Computer Vision, Freibourg, Germany, pp. 410–425.

Nevatia, R. and Huertas, A., 1998. Knowledge-based building detection and description: 1997-1998. In: Proceedings of the ARPA Image Understanding Workshop, Monterey, California.

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

Rousseeuw, P. and Leroy, A., 1987. Robust Regression and Outlier Detection. John Wiley & Sons, New York.

Stewart, C. V., 1999. Robust parameter estimation in computer vision. SIAM Reviews 41(3), pp. 513–537.

Stilla, U., Michaelsen, E. and Lütjen, K., 1996. Automatic extraction of buildings from aerial images. IAPR Workshop graz austria.

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

Torr, P. H. S., 1999. Model selection for structure and motion recovery from multiple images. Technical Report MSR-TR-99-16, Microsoft.

Vestri, C. and Devernay, F., 2000. Improving correlation-based dems by image warping and facade correlation. In: Proceedings of the International Conference on Computer Vision and Pattern Recognition.

Weidner, U. and Förstner, W., 1995. Toward automatic building extraction from high resolution digital elevation models. In: ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 50number 4, pp. 38–49.

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