Mapping by Dragging and Fitting of Wire-Frame Models

George Vosselman and Henri Veldhuis.

Abstract

Semi-automatic measurement of objects with regular shapes can be performed efficiently in three steps: (1) selection of an object model and approximate alignment of its wire frame by an image analyst, (2) precise alignment to the image by a fitting algorithm, and (3) correction of fitting errors, again by the image analyst. This paper presents a new approach to perform these three steps using the same principle in all three steps. The developed approach allows the image analyst to drag both points and lines of the projected wire frame, including curved edges and contour edges, in order to align these features with the image contents. Using the described algorithm, there is no need for the image analyst to specify which parameters of the object models are to be adapted in order to improve the alignment. The performance of the fitting step is analyzed and compared with an alternative approach.

Introduction

Future (geographic) information systems will contain three-dimensional (3D) and highly structured information. The development of procedures for the extraction of 3D object models from digital imagery is, therefore, receiving much attention at research institutes. Whereas efforts to fully automate the process of building extraction show good progress (Henricsson and Baltsavias, 1997; Steinhage, 1997), it is clear that under many circumstances automation is extremely difficult to achieve. For this reason, semi-automatic approaches are being developed which allow interaction between computer algorithms and an image analyst, thus taking advantage of the excellent interpretation and evaluation skills of the image analyst.

The measurement of objects can be dealt with most efficiently if the objects can be modeled by a combination of a few primitive shapes. These composite object models can be constructed prior to the measurement process. The model-based semi-automatic measurement of objects can then be seen as a three-step procedure: (1) the selection of an object model from a library and the approximate alignment of the object model with the object appearance in the images by an image analyst, (2) the precise alignment of the object model to the image by a fitting algorithm, and (3) the correction of erroneous fitting results, again by an image analyst.

In this paper we will present a new approach in which we use the same principle for modifying the pose and shape of the object models in all three steps.

The next section first reviews some of the related work presented in the last few years. In sections three to five the three steps of the measurement approach will be elaborated. The fitting algorithm is described in section three. This algorithm is similar to the one described by Lowe (1991). Success rates, precision, speed, and pull-in range are analyzed in a mapping experiment involving buildings in a residential and an industrial area. The results are compared with those obtained with the snake-like algorithm described by Fua (1996). In section four, the designed algorithm is applied to a different set of observations such that it becomes suitable for correcting erroneous fitting results. Section five demonstrates that the same algorithm can also be used for approximate alignment of the object models by an image analyst. The main features of the developed methods are summarized in the final section.

Related Work

Representation of Object Models

Object models can be described in various ways (Mortenson, 1997). The two most important types are briefly discussed, because the way of representing an object model can have an impact on the way in which the parameters of the model can be manipulated.

Constructive Solid Geometry (CSG) is widely used for computer aided design (CAD). With this technique an object is composed by taking unions and intersections of several primitive shapes such as rectangular boxes, spheres, cylinders, cones, and tetrahedra. Such an object model is described by specifying for each primitive the values of the shape parameters and the six pose parameters. Often the absolute pose parameters are specified for one primitive only and the pose parameters of the other primitives are described relative to the first primitive.

Boundary representations (B-rep) of objects describe the geometry of the points, edge lines, and surfaces of the object boundaries together with the topological relationships between these points, lines, and surfaces. A polyhedral object can therefore be described by the coordinates of the corner points and the point numbers that make up the lines and faces.

Manipulation of Wire Frames

The functions available in CAD packages to change the position and shape of object models are often fairly limited. Common approaches are the editing of numeric values of the object parameters and slide bars for rotations around a specified axis. These tools are primarily designed for the construction of CAD models. Whereas they can be used for approximate alignment of an object model with an image, they are not very user friendly

Delft University of Technology, Faculty of Civil Engineering and Geosciences, Thijsseweg 11, 2629 JA Delft, The Netherlands (g.vosselman@geo.tudelft.nl).

H. Veldhuis is currently with the Grontmij Geogroep, Bovendonk 29, 4700 BS Roosendaal, The Netherlands (Henri.Veldhuis@grontmij.nl).

Photogrammetric Engineering & Remote Sensing
Vol. 65, No. 7, July 1999, pp. 769–776.

0099-1112/99/6507–769$3.00/0

for the purpose of model-based measurement. Therefore, new tools need to be developed.

An approach dedicated to measurement with parameterized models is described by Lang and Förstner (1996). By dragging corner points of an object model to the corresponding position in an image, one or two parameters of the object model can be adapted. Which parameters are adapted depends on the chosen reference point, the point that is moved, and the entries of a so-called association table. A table has to be made for each object model available in the object model library. The tables can be edited in order to store the preferences of the image analyst.

Fitting Algorithms

After the approximate alignment of the object model by an image analyst, more precise estimates of pose and shape parameters can be obtained by fitting algorithms. Several approaches to optimize the alignment of an object model can be found in the literature.

Lowe (1991) describes a least-squares algorithm that fits the edges of the projected wire frame to edge pixels. These edge pixels are pixels with a grey value gradient above some pre-set threshold. Starting with an approximate alignment, the errors in the alignment are quantified by the perpendicular distances of the edge pixels to the nearest edge of the wire frame. The task of the fitting algorithm is to estimate the changes to the values of the pose and shape parameters that have to be applied in order to minimize the square sum of these distances. This is accomplished by setting up the observation equation

$$E\{\Delta u\} = \sum_{i=1}^{i=K} \frac{\partial u}{\partial p_i} \Delta p_i \qquad (1)$$

for each edge pixel. In this equation Δu is the perpendicular distance of an edge pixel to the nearest edge of the wire frame, p_i are the object parameters, and K is the number of parameters. An iterative least-squares adjustment results in an optimal estimation of the object parameter values. The required partial derivatives are obtained by numerical simulation.

Sester and Förstner (1989) describe a clustering algorithm followed by a robust estimation of pose and shape parameters. Both algorithms are based on correspondences between edges of the wire frame model and straight edges that are extracted from the image. Whereas the clustering algorithm is not suitable for the simultaneous determination of a large number of parameter values, the robust estimation is applicable to measurement of polyhedral objects.

Fua (1996) uses a snake-like approach to fit a polyhedral object model to an image. An energy function is defined as the negative sum of the absolute grey value gradients along the edges of a projected wire frame. With a steepest gradient algorithm the positions of the corners of the polyhedral model are adapted such that the value of the energy function is minimized. Instead of the typical regularization component in the energy function used for road mapping, Fua employs constraints on the direction in which the coordinates of the object corners may be adapted. These constraints are necessary to ensure that the geometric relationships between perpendicular and parallel lines of the object model are maintained.

Further Automation in Mapping Buildings

Fitting methods require approximate values of the object parameters. The derivation of an approximately aligned object model can be done completely manually, but, in order to further speed up the mapping process, (partial) automation is required. Gülch *et al.* (1998) present an approach for the measurement of buildings with gable roofs which usually only requires the image analyst to point out both ridge points in one image. Based on this knowledge, a search and matching process is started to find additional roof edges and reconstruct the 3D building model. A very similar approach is presented by Li *et al.* (1998). In their paper, the number of manual measurements ranges from one to three for flat roofs and two to four for gable roofs. The number of measured points per model depends on the success of the hypothesis generation conducted after each measured point. Hsieh (1996) developed another interactive modeling environment in which user-delineated roof boundaries in one image are used as cues in a matching algorithm to determine the ground level and roof level.

Due to the complexity of aerial image understanding, the automation processes described above are commonly restricted to simple building types such as flat roof or gable roof buildings. In this paper we will not focus on a high degree of automation for the measurements of a few specific models, but, rather, pursue the development of efficient tools for the measurement of the large class of arbitrarily shaped parameterized models.

Fitting Object Models to Images

In this section, we first outline some drawbacks of the described fitting methods. We then show how these drawbacks can be eliminated and describe several implementation aspects. In the last paragraph the performance characteristics of the new fitting procedure are analyzed and compared with those of the snake-like approach by Fua (1996).

Analysis of Fitting Methods

The purpose of a fitting algorithm is to determine the pose and shape parameters of an object model such that the edges of the wire frame, as projected into the image(s), are optimally aligned with the pixels with high gradients. It is, of course, assumed that these pixels correspond to the projection of the object's edges. This objective is most clearly visible in the energy function of the snakes approach by Fua (1996). However, the optimization algorithm used to find the best parameters is computationally expensive, because the derived energy gradients only indicate the direction in which the vector of parameter values should be changed. It takes a substantial number of iterations to accurately determine the amount of change required to obtain the best fit.

Computationally, the least-squares approaches by Lowe (1991) and Sester and Förstner (1989) are much faster. Due to the edge detection, the direct relationship between parameter value changes and gradient values is, however, lost. Weak edge pixels will not be used in the parameter estimation if their gradients are below the threshold for edge detection.

Modification of Lowe's Fitting Method

A direct relationship between the estimation of the pose and shape parameters and the gradients of the pixels can be achieved by modifying the approach by Lowe (1991). Lowe introduces an observation equation for each pixel which fulfils two conditions: (1) the grey value gradient should be above some threshold and (2) the pixel should be within some range of a projected wire frame edge. We propose to drop the first condition: i.e., for all pixels within some range of a wire frame edge, the observation equation is introduced. In order to ensure that the pixels with the higher gradients dominate the parameter estimation, the squared grey value gradients of the pixels are used as weights to the observation equations:* i.e.,

*This is similar to the estimation of subpixel coordinates for points found with the interest operator described in Förstner and Gülch (1987). In this paper the coordinates of all pixels within a selected window are used to estimate the interest point by using the squared grey value gradients as weights.

$$E\{\Delta u\} = \sum_{i=1}^{i=K} \frac{\partial u}{\partial p_i} \Delta p_i, \qquad W\{\Delta u\} = \left(\frac{\partial g}{\partial u}\right)^2. \tag{2}$$

In this way, we avoid the problem of selecting a threshold for edge detection and at the same time improve the accuracy of the fitting because the stronger edges will have more weight in the estimation whereas all edge pixels have unit weight in Lowe's approach.

The grey value gradient used in the weight function is calculated as the partial derivative of the grey value g in the direction u perpendicular to the edge of the wire frame. This has the advantage that gradients caused by background objects with perpendicular directions will not interfere with the parameter estimation.

Implementation Aspects

Gradient pixels of other objects will disturb the parameter estimates if they are within some distance of the wire frame edges of the object to be measured. In order to reduce the bias by other objects, it would be best to only use pixels within a very small buffer around the edges. Using a small buffer, however, implies that the image analyst needs to supply an accurate approximate positioning of the wire frame. This would reduce the benefits of a fitting method. It is therefore better to start with a large buffer and then to reduce the buffer size after each iteration of the least-squares estimation.

A large buffer will contain a large number of pixels. Setting up an observation equation for each pixel would make the fitting algorithm time consuming. In the first few iterations with a larger buffer size, we therefore subsample the buffer with profiles perpendicular to the edges (see Figure 1). Observation equations are set up only for points on these profiles. The point density is increased each time the buffer size is reduced. In this way a large initial buffer size can be combined with a fast and precise fitting.

The partial derivatives of the distances with respect to changes in the parameters are estimated numerically, as in Lowe's approach. A more detailed explanation of the estimation can be found in Vosselman (1998).

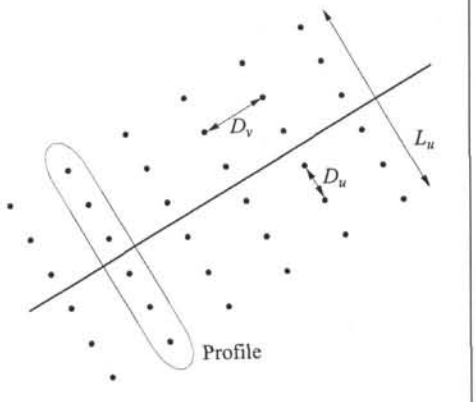


Figure 1. Profiles perpendicular to a projected wire frame edge. The point density can be controlled by the distance between the profiles D_v, the length of the profiles L_u, and the number of points on a profile N_u. The distance between points within a profile is defined by $D_u = L_u/N_u$.

Performance Analysis

To assess the performance of the designed fitting algorithm and to compare it with the snake-like algorithm by Fua (1996), several experiments have been performed. The imagery and reference data were taken from the Avenches data set provided for the first Ascona Workshop (Gruen *et al.*, 1995). The images were recorded at a scale of 1:5000 and scanned with a pixel size of 15 μm (7.5 cm on the ground).

After a description of the used object models the set-up of the experiments is outlined. Finally, the performance of both fitting algorithms is analyzed with respect to success rates, speed, precision, and pull-in range.

Roof Models

All measurements were performed with a standard gable roof model. This roof model can be described by seven parameters: the three coordinates of a roof point (X, Y, Z); the rotation around the Z-axis (κ); and the length (l), width (w), and height (h) of the roof. The height is defined as the height difference between the roof ridge and the gutter. Note that the wall height is not estimated in the fitting experiments.

The snake-like approach used a polyhedral boundary representation. In this model the three coordinates of all six roof points are treated as unknown parameters. To make sure that the coordinates estimated by the fitting algorithm still form a gable roof, eleven shape constraints are used (Veldhuis, 1998).

Experimental Protocol

A total of ten buildings with gable roofs was selected from two stereo pairs. For all buildings, the wire frame of the roof model was approximately positioned by an image analyst. Using these approximate positions, 15 measurements were performed with both fitting methods for each building. These different measurements were used to study the effects of the subsampling of the buffer onto the success rates and the computation time of the fitting. The subsampling is used for both the least-squares and the snakes approach. For the distances between the profiles D_v and the initial profile length L_u, 5, 10, or 20 pixels were used. The initial number of points on a profile N_u was either 5 or 10. All combinations were made, however, with the restriction that the minimum distance between points in a profile is one pixel. Therefore, $L_u = 5$ was not combined with $N_u = 10$. In the last iteration, the number of points in a profile is set to three and the distance between the points is set to one pixel.

The analyses of the precision and the pull-in range were done with the subsampling which resulted in the highest success rates. For the analysis of the pull-in range, another set of experiments was performed in which the errors in the approximate position and rotation of the roof models were systematically increased until the fitting algorithm could no longer make a correct estimate.

Success Rates

Looking at the fitting results, it was found that many of the fitted models were not fully aligned. Often one or two edges were wrong due to the presence of other objects in the image. Most of the edges, however, were positioned correctly. The success rate was therefore defined as the percentage of wire frame edges that required no correction by the image analyst. The results are visualized in Figure 2. For many combinations of subsampling parameters, both methods achieved success rates over 90 percent. The least-squares approach seems to be a little more robust with respect to changes in the subsampling. Both methods show a lower success rate with an initial profile length of only 5 pixels. For both methods, best results were obtained with an initial profile length of 10 pixels, 5 points per profile, and a distance between the profiles of 5 pixels.

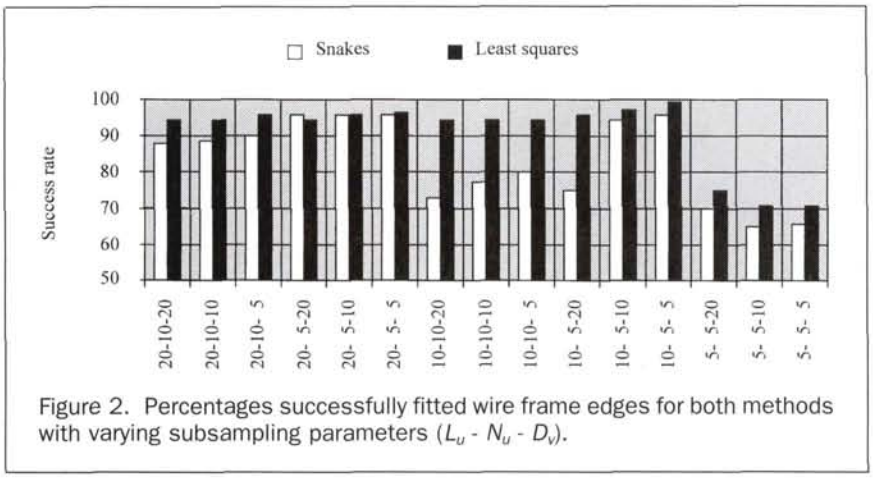


Figure 2. Percentages successfully fitted wire frame edges for both methods with varying subsampling parameters (L_u - N_u - D_v).

Speed

The results on the CPU-time used for fitting are shown in Figure 3. It is clear that the least-squares method is much faster than the snake-like method . This is due to the comparatively large number of iterations required by the last method. Also clearly visible is the dependency of the computation time on the number of points for which observation equations are derived.

Another critical factor with respect to the computation time is the hidden line analysis. The performance of hidden line algorithms is no bottleneck when working with polyhedral objects but does become one when working with curved objects such as cylinders (Ermes and van den Heuvel, 1998).

Precision

For the evaluation of the precision, standard deviations were computed for both the coordinates of the roof corners (Table 1) and the pose and shape parameters (Table 2). Both fitting methods obtain approximately the same results. The new least-squares fitting method is often slightly better, but the

TABLE 1. STANDARD DEVIATIONS OF THE ESTIMATED ROOF CORNER COORDINATES.

		X	Y	Z
Least squares	(m)	0.09	0.09	0.14
	(pixel)	1.2	1.2	1.9
Snakes	(m)	0.12	0.10	0.12
	(pixel)	1.6	1.4	1.6

TABLE 2. STANDARD DEVIATIONS OF THE ESTIMATED POSE AND SHAPE PARAMETERS. THE STANDARD DEVIATION OF THE κ-ROTATION IS EXPRESSED IN DEGREES, ALL OTHER VALUES ARE IN METERS OR PIXELS.

		X	Y	Z	κ	l	w	h
Least squares	(m)	0.06	0.05	0.07	0.22	0.06	0.09	0.09
	(pixel)	0.8	0.7	1.0	0.22	0.8	1.2	1.2
Snakes	(m)	0.09	0.07	0.12	0.22	0.08	0.07	0.09
	(pixel)	1.2	1.0	1.6	0.22	1.0	0.9	1.3

Z-coordinate of the roof points and the house width were better estimated by the snake-like approach.

Considering the expected accuracy of the reference data (0.1 m), it is difficult to assess the accuracy potential of the fitting methods. It can, however, be concluded that the reference data probably had a standard deviation below 0.1 m and that both fitting methods are capable of obtaining the same order of precision.

Pull-In Range

An important reason for using fitting techniques is the reduction in mapping time. The image analyst can work faster if the object models only need to be positioned approximately. Therefore, the benefits of a fitting method increase when the estimated parameters are correct even though the approximate positioning was bad. The pull-in range is defined as the range of differences between the correct and approximate parameter values for which the measurement still succeeds. This pull-in

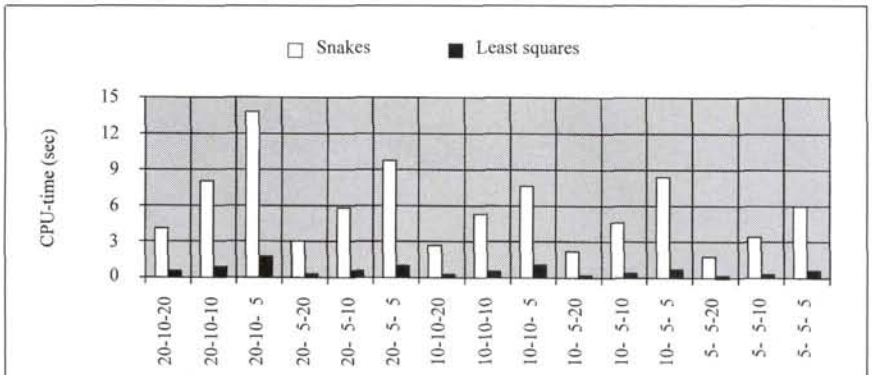


Figure 3. Fitting speed for both methods with varying subsampling parameters (L_u - N_u - D_v).

range was investigated for a shift of the object model in the *XY*-direction (Figure 4a) and a rotation around the *Z*-axis (Figure 4b) by systematically increasing the errors in the approximate positioning until the fitting failed. The experiment was performed for all ten buildings. The results show that the pull-in range for the least-squares method is a bit larger than for the snake-like emethod.

Of course, the pull-in range will also depend on the size of the buffers around the wire frame edges and the presence of other object edges. The subsampling parameters were, however, not varied in this experiment.

Correction of Fitting Results

Errors in Fitting Results

Results of fitting algorithms usually contain errors. However, often the majority of the model edges will be correctly aligned by a fitting algorithm, whereas only a few lines are misaligned due to, for example, interfering neighboring objects.

Figure 5 shows a typical example of building extraction from aerial imagery. The two images on the left side are patches of the two photographs of a stereo pair overlaid with an approximately aligned wire frame of the house model. The two images in the second column show the result of the fitting algorithm. Most edges are aligned correctly, but the high contrast between the house's shadow and the yard attracted the lower horizontal edge in the first image. The correct position would have been a few rows higher. Due to this mismatch, the sloped lines of the roof in the second image also show errors. The slope is underestimated.

Objective for Correction of Fitting Errors

Because most of the edges are aligned correctly, measurements by the image analyst only have to deal with one or more incorrectly aligned edges. However, in a parameterized object model, the position of edges can not be adapted individually. Instead, we need to find the changes to the values of the pose and shape parameters that correct the errors.

For an image analyst, this may be a difficult task to perform. If only one parameter needs to be changed, this can be done fairly easily. However, often a combination of two or more parameters needs to be changed to obtain the correct alignment. It is not trivial to find this combination without having to redo the complete measurement by hand.

In the correction method described in the next paragraph, we therefore only require the image analyst to measure a few points on one (or a few) of the edges in the image that were not found correctly. Now, the objective of the correction method is to adjust the values of the pose and shape parameters such that the wire frame edges coincide with the points supplied by the image analyst and at the same time to keep the good fitting results as much as possible.

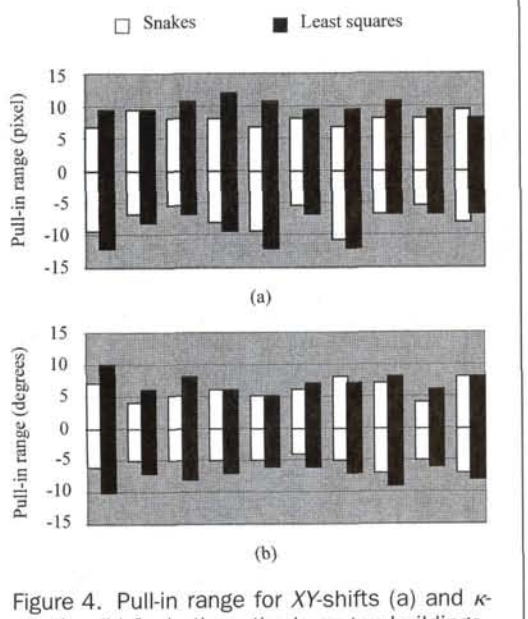


Figure 4. Pull-in range for *XY*-shifts (a) and κ-rotation (b) for both methods on ten buildings.

Algorithm for Correction of Fitting Errors

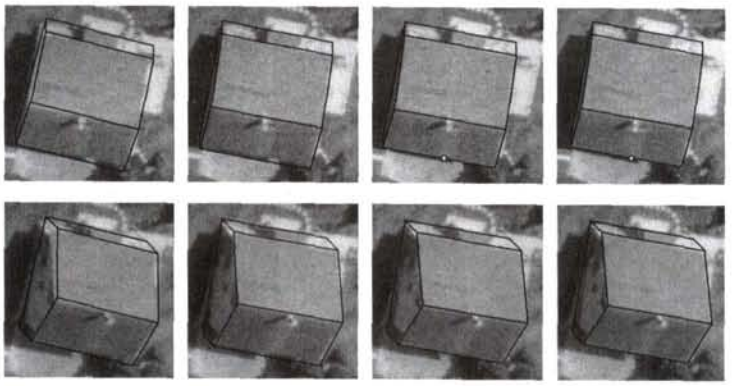
The above objective can be met with the same principle as used for the fitting algorithm. By measuring a point in the image, the image analyst specifies the correct position of a wire frame edge and implicitly also indicates that the nearest edge was aligned incorrectly. A better estimation of the pose and shape parameters can now be found by removing from the normal equation system the observation equations of the pixels within the buffer of the indicated edge and by adding an observation equation for the point measured by the image analyst. In order to ensure that the adjusted wire frame edge will go through the measured point, this observation equation is given a high weight. Thus, there are two types of observation equations: equations for the pixels within the buffers of the correctly fitted edges and equations for the points measured by the image analyst. Due to the geometric constraints within parameterized models, the image analyst usually only will have to measure one point per incorrectly fitted edge. Hence, the correction of fitting errors can be done very efficiently.

In the third column of Figure 5, the results of the correction by a single point measurement are shown. The model now fits quite well to the image. As can be seen from the parameter values below the images, the main corrections have been applied to the roof height and the width of the house.

Alternatively, the correction can also be done in a slightly less optimal but computationally much faster way. In this approach, we do not make use of observation equations for the pixels in the buffers of the correctly fitted edges. Instead, these edges are sampled with a number of points (see Figure 6). For each of these points, an observation equation is set up with a unit weight. The observation equations derived from the points measured by the image analyst are again added with a very high weight. The resulting adjusted object model will therefore accurately be aligned with the measured points, whereas the other points ensure that the location of the correctly fitted edges changes as little as possible.

This last correction method, however, does not completely eliminate the effects of the edge mismatch. Because the positions of the wire frame edges are correlated due to the shape constraints in the object model, an error in the fitting of one edge will also disturb the location of other edges. This was noticeable in the second picture in the second row of Figure 5. If only one point to correct the lower horizontal edge is measured, the algorithm will try to maintain the position of the edges with the incorrect slope. The result of a correction with this method can be seen in the images in the last column of Figure 5. Due to the correlation between the mismatched edge in the first image and the slope edges in the second image, the measurement in the first image largely corrects the errors in the second image too. Therefore the differences between the two correction methods are very small.

Whereas the first method gives a better correction, the second method is much faster because only a small number of observation equations are involved. Thus, the correction becomes fast enough to enable the image analyst to drag the wire frame into the correct position. The parameter estimation



Approximate values		Fitting results		Optimal correction		Approximate correction	
X	569852.32	X	569852.60	X	569852.66	X	569852.61
Y	191610.30	Y	191610.05	Y	191610.07	Y	191610.00
Z	481.33	Z	481.54	Z	481.61	Z	481.56
κ	64.12	κ	59.51	κ	59.62	κ	59.22
length	11.87	length	12.00	length	11.93	length	11.96
width	12.55	width	12.69	width	12.34	width	12.34
roof height	2.69	roof height	2.91	roof height	3.27	roof height	3.28

Figure 5. Results of approximate alignment, fitting, and corrections for a house model simultaneously aligned in two images. All coordinates and sizes are in meters, the rotation angle κ is in degrees. The measured point for the correction is indicated by the white dot.

can be done within the loop which processes the mouse motion events without noticeably delaying the mouse motion.

The edges that have been fitted correctly are sampled with a constant interval. This has the effect that longer edges will get a higher weight in the adjustment. Of course, for straight edges, two points with a weight depending on the edge length would have been sufficient. For curved edges, and therefore in general, it is however easier to work with a standard sample interval and a unit weight.

With the *second correction method*, the image analyst can drag wire frame edges to the correct position. It is, however, also possible to do the same with distinct points of the wire frame, such as corners of a polyhedral object. Measurement of such points then results in two observation equations per point, related to the row and the column coordinate in the image: i.e.,

$$E\{\Delta r\} = \sum_{i=1}^{i=K} \frac{\partial r}{\partial p_i} \Delta p_i \tag{3}$$

$$E\{\Delta c\} = \sum_{i=1}^{i=K} \frac{\partial c}{\partial p_i} \Delta p_i \tag{4}$$

Here, Δr and Δc are the coordinate differences between the measured point and the current position of the wire frame corner in the image. Thus, the image analyst can modify the pose and shape parameters by dragging both points and edges of the wire frame.

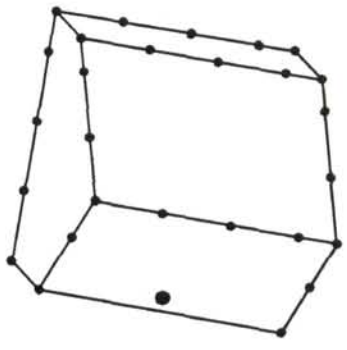
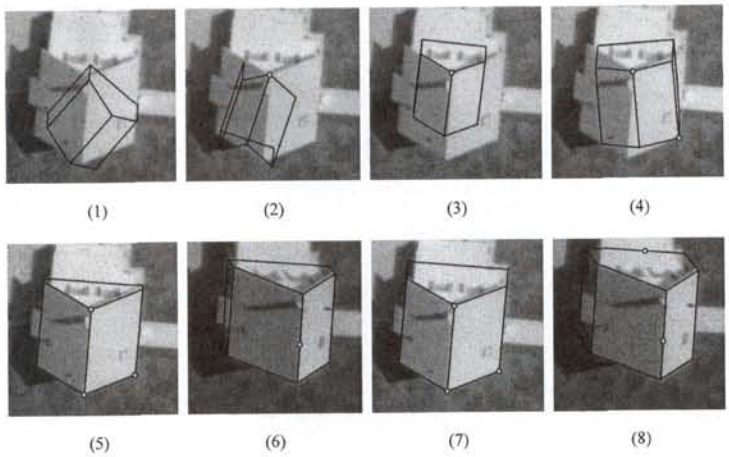


Figure 6. Correctly fitted edges are sampled with points (small dots). One measurement (large dot) per incorrectly fitted edge is often sufficient to correct the error.

Approximate Alignment of Object Models

The principle of the second correction method can also be used for the approximate positioning of a wire frame model in the images. The behavior of the algorithm is demonstrated in Figure 7. The first image shows the initial position of the wire frame. The second image shows the result after one ridge point is dragged to the correct position. This is done by using Equations 3 and 4 for the ridge point with a high weight and Equation 1 for the points on the sampled edges with a low weight. Because the algorithm tries to keep the edges at the same location, the shape parameters may take strange values (see, for example, the negative roof height) and the model becomes difficult to interpret. Although one could proceed the measurement by dragging other points to their correct positions,



Parameters	Image 1	Image 2	Image 3	Image 4	Image 5	Image 6	Images 7 and 8
X	569855.39	569860.67	569861.40	569861.05	569862.05	569863.52	569863.52
Y	191677.00	191675.54	191674.32	191675.04	191673.42	191671.43	191671.43
Z	468.38	467.84	470.08	468.73	471.95	475.89	475.89
κ	0.00	-22.96	-33.44	-45.64	-36.14	-36.14	-36.14
length	8.00	9.22	8.00	9.69	10.72	10.66	10.66
width	8.00	6.89	8.00	9.55	12.45	12.39	12.39
roof height	3.00	-2.90	3.00	1.32	4.75	4.72	4.72
wall height	3.00	3.97	3.00	4.33	0.76	1.52	3.45

Figure 7. Manual alignment of the object model to two images of a stereo pair (see text). The wall height is the distance between the ground and the roof gutter.

working with these kind of models is clearly not very user friendly.

This problem can be easily solved by first adapting the pose parameters without changing the shape parameters. By removing the shape parameters as unknowns from the observation equations, the algorithm will try to fit the model to the measured point and the sample points on the edges by modifying the pose parameters only. The resulting house model is shown in the third image of Figure 7.

Now, one point of the model is at the correct position. If we drag a second point to its correct position in the same way as the first point, this first point will not remain at the same position. To ensure that points that have been dragged to their correct position are not moved, Equations 3 and 4 are added for each measured corner point. This is indicated by the white dots in the images. Because the pose of the house model is approximately correct, the shape parameters were no longer fixed during the measurement of the second point. The order in which the remaining points are being measured is not important. In the fourth image of Figure 7, a gutter point is dragged to its position. The fifth image shows how the second ridge point is measured. In order to determine the height of the house, the roof ridge is also measured in the overlapping photograph as shown in the sixth image. Note that the point measured in this image is related to an edge (instead of a wire frame node) and therefore results in only one equation. Any point on an edge in the second image could have been used for determining the height as long as the selected edge is not on an epipolar line. Finally, another edge is measured to adapt the wall height. The last two images show the model at the end of the measurement process.

Compared to the manual alignment method described by Lang and Förstner (1996), there are several advantages:

- The image analyst is able to drag points as well as lines of the wire frame and thus has more ways to change the model parameters of the object.
- These object lines may also be contour lines (lines that delineate the contour of an object part in an image but do not necessarily represent a physical edge of the three-dimensional object). This feature makes the alignment method suitable for curved objects too.
- The points of the wire frame nodes are not associated with one or two object parameters. As can be seen in Figure 7, the measurement of an object point usually involves changes in a large number of pose and shape parameters. The absence of a direct relationship between points of the wire frame and parameters of the object model has two advantages. First, there is no need for association tables. Second, the image analyst does not need to have knowledge about the object parameterization. Therefore, the analyst does not need to reason which parameters are to be changed in order to find the best alignment.

Conclusions

In this paper we presented a new approach for interactive alignment of parameterized object models to images. The same principle for adapting the pose and shape parameters of the object

models has been used in all three measurement phases: approximate alignment, fitting to grey value gradients, and correction of fitting errors.

The method to manually align the model to the images offers a great amount of flexibility. Points, object edges, and contour lines can be dragged to their correct location, whereas the image analyst does not need to know which parameters have to be changed for this purpose.

For the fitting algorithm, a direct relationship between the required adaptation of the pose and shape parameters and the grey value gradients of the pixels has been formulated. The estimation shows fast convergence and does not need thresholds for the determination of edge pixels. The mapping accuracy is better than 10 cm for 1:5,000-scale imagery scanned with a 15-μm pixel size.

Finally, two methods have been designed to modify the object parameters in case of fitting errors. These errors can be eliminated efficiently by combining observation equations for pixels along correctly aligned wire frame edges with equations for corrective measurements by the image analyst. Due to shape constraints within parameterized models, the number of required measurements is often very small. A non-optimal but very fast correction algorithm showed only very small deviations from the optimal estimates.

Acknowledgment

This research is supported by the Dutch Technology Foundation (STW).

References

Ermes, P., and F.A. van den Heuvel, 1998. Measurement of Piping Installations by Digital Photogrammetry, *International Archives of Photogrammetry and Remote Sensing*, 32(Part 5):217–220.

Förstner, W., and E. Gülch, 1987. A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features, *Proceedings of the ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, Interlaken, Switzerland, pp. 281–305.

Fua, P., 1996. Model-Based Optimization: Accurate and Consistent Site Modeling, *International Archives of Photogrammetry and Remote Sensing*, 31 (Part B3):222–233.

Grün, A., O. Kübler, and P. Agouris (editors), 1995. *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Birkhäuser Verlag, Basel, 321 p. (information about the Avenches dataset can be found at **http://www.p.igp.ethz.ch/p02/projects/AMOBE/dataset-new.html**).

Gülch, E., H. Müller, T. Labe, and L. Ragia, 1998. On the Performance of Semi-Automatic Building Extraction, *International Archives of Photogrammetry and Remote Sensing*, 32 (Part 3/1):331–338.

Henricsson, O., and E. Baltsavias, 1997. 3-D Building Reconstruction with ARUBA: A Qualitative and Quantitative Evaluation, *Proceedings Second Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Ascona, 4–9 May, Birkhäuser Verlag, pp. 65–76.

Hsieh, Y., 1996. SiteCity: A Semi-Automated Site Modelling System, *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, California, 18–20 June, pp. 499–506.

Lang, F., and W. Förstner, 1996. 3D-City Modeling with a Digital One-Eye Stereo System, *International Archives of Photogrammetry and Remote Sensing*, 31 (Part B3):415–420.

Li, J., R. Nevatia, and S. Nornoha, 1998. User Assisted Modeling of Buildings, *Proceedings Image Understanding Workshop*, Monterey, California, 20–23 November, pp. 571–576.

Lowe, D., 1991. Fitting Parameterized Three-Dimensional Models to Images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441–450.

Mortenson, M.E., 1997. *Geometric Modeling*, John Wiley & Sons, New York, 523 p.

Sester, M., and W. Förstner, 1989. Object Location Based on Uncertain Models, *Mustererkennung 1989*, Informatik Fachberichte 219, Springer Verlag, pp. 457–464.

Steinhage, V., 1997. On the Integration of Object Modeling and Image Modeling in Automated Building Extraction from Aerial Images, *Proceedings Second Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Ascona, 4–9 May, Birkhäuser, pp. 139–148.

Veldhuis, H., 1998. Performance Analysis of Two Fitting Algorithms for the Measurement of Parameterised Objects, *International Archives of Photogrammetry and Remote Sensing*, 32 (Part 3/1):400–406.

Vosselman, G., 1998. Interactive Alignment of Parameterised Object Models to Images, *International Archives of Photogrammetry and Remote Sensing*, 32 (Part 3/1):272–278.