Efficient 3D-modeling of buildings using a priori geometric object information

Frank A. van den Heuvel, George Vosselman[1]

Delft University of Technology
Department of Geodetic Engineering
Thijsseweg 11, 2629JA Delft, The Netherlands

ABSTRACT

The subject of this paper is the research that aims at efficiency improvement of acquisition of 3D building models from digital images for Computer Aided Architectural Design (CAAD). The results do not only apply to CAAD, but to all applications where polyhedral objects are involved. The research is concentrated on the integration of a priori geometric object information in the modeling process. Parallelism and perpendicularity are examples of the a priori information to be used. This information leads to geometric constraints in the mathematical model. This model can be formulated using condition equations with observations only. The advantage is that the adjustment does not include object parameters and the geometric constraints can be incorporated in the model sequentially. As with the use of observation equations statistical testing can be applied to verify the constraints. For the initial values of orientation parameters of the images we use a direct solution based on a priori object information as well. For this method only two sets of (coplanar) parallel lines in object space are required.
The paper concentrates on the mathematical model with image lines as the main type of observations. Advantages as well as disadvantages of a mathematical model with only condition equations are discussed. The parametrization of the object model plays a major role in this discussion.

Keywords: close-range photogrammetry, computer vision, line photogrammetry, geometric modeling, topology, geometric constraints, mathematical model, sequential adjustment, object reconstruction, CAAD

1. INTRODUCTION

When acquiring 3D computer models of buildings we have to be aware of the requirements set by the user of these models. A first distinction can be made between so-called city models and architectural models. The first type of model usually covers (a part of) a city and contains relatively simple house models. Often aerial images are used which leads to an emphasis on modeling of the roofs of the buildings. As there is an increasing demand for city models the amount of research on the efficient acquisition of these models is considerable.[2-4, 8] The second type of model generally contains only one building or a single block of buildings and will usually show detail in the shape of the façades. Now close-range imagery is needed and the models are more complex. The increase in complexity of the building model is the reason why we do not want to work with only a limited number of predefined basic shapes called primitives (see section 3). These primitives are often used to build the model in the approaches for the acquisition of city models. Here the idea is that the operator can construct new primitives in interaction with the images and then reuses these shapes in order to increase efficiency. Predefined primitives with a parametric representation are used in other approaches for the acquisition of architectural models as well.[1, 9] In our approach a shape is represented by topological relations and geometric constraints between object features (points, lines and planes) because we can not expect the operator to supply a parametric representation of the new shape without constraints. The mathematical model we adopted is based on condition equations with observations and does not contain parameters apart from image pose parameters in case the position and orientation of the images are only approximately known (section 4). The main advantage of a mathematical model with only conditions on the observations is that there is no need for parametrization of the object (or primitive) and as a result there is no need for the computation of approximate values for these parameters. Furthermore the data supplied by the operator can be processed sequentially (see section 6) resulting in an early detection of possible errors. Disadvantages are the formulation and the choice of (independent) condition equations (section 5) and the fact

[1] E-mail: F.A.vandenHeuvel@geo.tudelft.nl , M.G.Vosselman@geo.tudelft.nl

Reference:
F.A. van den Heuvel, M.G. Vosselman, 1997. Efficient 3D modeling of buildings using a priori geometric object information.
in Videometrics V, Sabry F. El-Hakim (ed.), Proceedings of SPIE Vol. 3174, pp. 38-49

that the object model has to be computed from the adjusted observations in a separate step (section 7). In the next section the starting-points of the research are elucidated and some basic choices are explained.

2. PREMISES AND APPROACH

The research focuses on close-range photogrammetry for buildings but we want the results to be applicable to all objects that can be represented as a polyhedral shape. This is true for most of the man-made objects like buildings. Another important feature of these polyhedral man-made objects is that they often show repetitive patterns. In the case of buildings a multitude of examples can be supplied such as vertical walls or identical windows. In the next section the geometric modeling of the polyhedral objects will be discussed.

The geometry of the image will depend on the camera. We assume interior orientation parameters to be known and as a result for each location in the image the corresponding direction in object space is known in the camera system.

The relation between the camera system and the object system is supplied by the exterior orientation parameters. The mathematical model is formulated in such a way that two approaches are possible. In the first approach the exterior orientation parameters are known approximately. To arrive at approximate values the procedure outlined in section 4.3 can be used. This method relies on a priori geometric object information in the form of parallel lines. In this case the exterior orientation parameters will have to be improved by adjustment. The second approach assumes the exterior orientation parameters to be known to a sufficient degree. Then the exterior orientation parameters can be eliminated from the mathematical model.

The system concepts presented here do not rely on an automated image interpretation or segmentation. As many others we believe that real-life imagery requires a semi-automated approach.[1-4] In fact it should be possible to perform the interpretation and the measurements of the images fully manual. But since efficiency improvement is our goal, automation of image analysis tasks is the aim for the long-term. The system presented here is designed for manual processing of the results of low-level feature extraction. The extracted image features can be straight lines or image points. The research effort in image feature extraction is considerable, see e.g. Ref. 5 and 6. Relations between the image features is not assumed to be present beforehand. The topology is to be supplied by the operator through image interpretation. Apart from the topology the operator has to measure missing image features and can supply geometric constraints such as parallel object lines or perpendicular object planes. Efficiency is definitely improved when the operator can make use of Constructive Solid Geometry (CSG).[7] Then the object representation is built from geometric primitives selected by the operator. A primitive is a relatively simple shape such as a box and allows many object topology relations and geometry constraints to be supplied in one go. But we do not want to be limited to the use of predefined primitives. The operator should be able to build new primitives by specifying topology and constraints. It is clear that the need for operator interaction with the imagery as well as with the 3D-model under construction demands for a seamless integration of photogrammetric image processing functionality and CAD-functionality within one graphical user interface.

3. GEOMETRIC MODELING

In this section the adopted geometric image-object model is discussed. An explicit distinction is made between topology and geometry. Furthermore the choice of parametrization of the geometric features is presented.

Although in the next section a mathematical model is formulated in which the parametrization of the object model does not play a role, we can not do without geometric modeling. A choice has to be made for the representation of the object model and thus for the data structure to be used. The need for a data structure does not only arise in the object reconstruction phase, but in the image interpretation phase as well. In the image interpretation phase the projection of a wireframe is to be superimposed on the images in order to visualize the established relations between the images and the object model.

As in many advanced CAD systems we adopt a hybrid geometric modeling approach in the sense that we want to combine a boundary representation (B-rep) and a CSG approach. The operator then can work in two modes. The first mode is the B-rep mode in which mode the operator can construct object features (points, lines or planes in space) by combining point and line features in the images. Geometric constraints can be added to these object features. In this mode solids can be constructed as a combination of planes (and thereby lines and points) in space. B-reps of solids have to be validated. The boundary of a valid solid must be closed, orientable, non-self-intersecting, bounding and connected.[10] These solids then can act as primitives in the second mode, i.e. the CSG mode. In this mode primitives can be combined using boolean operations resulting in a binary CSG tree where the leaf nodes represent the primitives and the branch nodes are the boolean operators (union, difference and intersection). CSG modeling will be supported by specialized software, a so-called solid modeling kernel. The CSG mode will increase productivity considerably, especially when the object shows many repeating elements. For instance for windows in buildings this will be true. It is important to note that in this approach the primitives constructed by the operator do not have a parametric representation. They are represented by their topology in combination with geometric constraints.

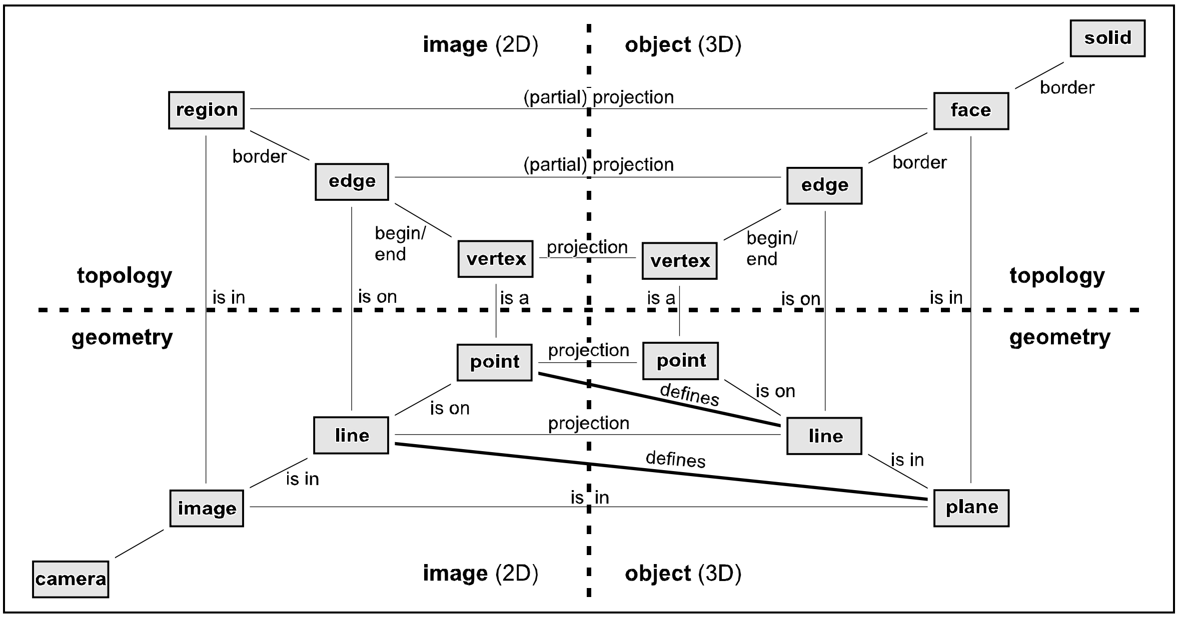

Figure 1: Geometric image-object model

The geometric model chosen (Figure 1) is based on the Formal Data Structure (FDS) presented in Ref.11. In this graph-based model there is a clear distinction between topology and geometry. This is important because the emphasis in the image interpretation process is on topology. The operator links the edges in the images to the edges in object space. Edges in object space are linked to faces and faces again to solids. On the geometry side of the geometric model the relations between the points and lines in the image and the points, lines and planes in object space are depicted. It has to be noted that lines and planes in object space can be features representing the object (then the *projection* relation holds) or features that relate image and object geometry (then the *defines* relation hold).

Table 1: Number of parameters of the entities of the geometric model.

entity	*image (2D)*	*object (3D)*	*remarks*
point	2	3	see Table 2
line	2	4	see Table 2
(image) plane	(6 : exterior orientation)	3	see Table 2
vertex	2	3	point parameters
edge	4	6	e.g. parameters of begin and end points
region / face	2 · (# edges)	3 + 2 · (# edges)	e.g. parameters of bordering lines (+ 3D-constraints)
solid	-	3 · (# faces)	e.g. parameters of bordering planes

The image itself and the camera that is used to take the image are added to the geometric model for completeness. The minimum number of parameters associated with each entity is listed in Table 1. The parametrization itself is chosen similar to the one used by Kanatani.[12] The parametrization of points, lines and planes is listed in Table 2.

Table 2: Parametrization of points, lines and planes in space

entity	*direction*	*position*	*constraints*
point		$\mathbf{x} = (x, y, z)$, $l_{\mathbf{x}}$	$\lvert\mathbf{x}\rvert = \sqrt{x^2 + y^2 + z^2} = 1$
line (image point)	$\mathbf{d} = (x, y, z)_{\mathbf{d}}$	$\mathbf{p} = (x, y, z)_{\mathbf{p}}$, $l_{\mathbf{p}}$	$\lvert\mathbf{d}\rvert = \lvert\mathbf{p}\rvert = 1$, $\mathbf{d} \cdot \mathbf{p} = 0$
plane (image line)	$\mathbf{n} = (x, y, z)_{\mathbf{n}}$	$l_{\mathbf{n}}$	$\lvert\mathbf{n}\rvert = 1$

To avoid the singularities which can arise when a line or a plane passes through the origin a direction vector of unit length is used in combination with a parameter (l) that specifies the distance to the origin of the coordinate system. This singularity does not occur for the position vector of a point. For uniformity a unit vector and a separate length parameter is used for the point parametrization as well. In this way the duality between point and plane representation is preserved. The parametrization is visualized in Figure 2. The same parametrization is used for the image features as well. Then only the *direction* (and *constraints*) column of Table 2 apply and $l_{\mathbf{p}}$ and $l_{\mathbf{n}}$ are zero for the representation of image features in the camera system.

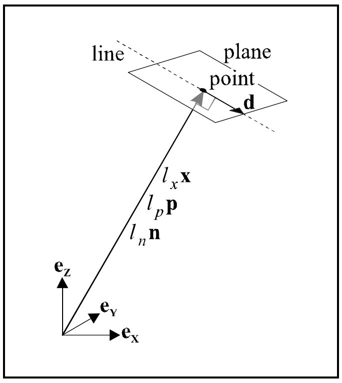

Figure 2: Parametrization of point, line and plane

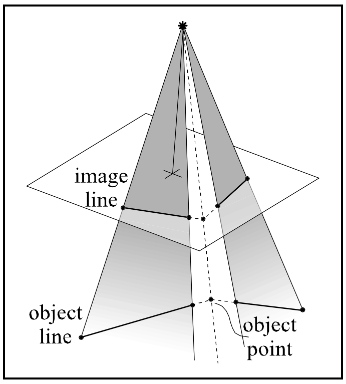

Figure 3: Image and object points and lines

4. A MATHEMATICAL MODEL FOR LINE PHOTOGRAMMETRY

In this section we present a mathematical model that differs from commonly adopted models in the sense that the model does not include object parameters. The exterior orientation parameters of the images can be eliminated from the model so a mathematical model results with condition equations with observations only. With this approach parametrization of the object does not play a role during the interactive modeling process. At the same time geometric object constraints can be examined and processed one by one. First the observation vectors are discussed and their transformation to the coordinate system of the exterior orientation parameters. Then the choice between parameters and constraints is made.

4.1 The Observations

The mathematical model aims at the adjustment of automatically extracted or manually measured straight line features in the image. The line features are the main type of observations. Lines in the image can be intersected resulting in image points (Figure 3). These points do not belong to the group of image point observations because they fully depend on the lines in the image that have priority in the line-photogrammetric approach presented here.

The image point observations are represented by the direction vector $\mathbf{d} = \frac{(x, y, -c)}{|(x, y, -c)|}$ where  is the location of the point in the camera system (relative to the principal point) and c is the focal length. The line features in the image all have a begin and end point, possibly on the border of the image. But these points do not have to correspond to vertices of the object. This is probably not the case if the lines result from automatic feature extraction and no segmentation is performed, i.e. the topology is unknown. This is the reason why each line feature is represented in the model by the normal vector $\mathbf{n}$ of the plane through the line and the projection center. If the begin and end point of the line are stored the plane in space (and thereby the corresponding line in object space) can be bordered if needed. The normal vector can be computed from the direction vectors of begin and end point through their cross product: $\mathbf{n} = \frac{\mathbf{d}^i \times \mathbf{d}^j}{\left|\mathbf{d}^i \times \mathbf{d}^j\right|}$. In principle only one type of observations exists namely the direction vector in space corresponding to a point or a line feature in the image, respectively a line or a plane in object space.

By using a three element direction vector we have over-parametrized the observations because a direction in space is defined by only two parameters. In relation, the covariance matrix of the observations is singular if the focal length is assumed to be a non-stochastic constant. For the point observations the focal length could be eliminated as an observed quantity but for the normal vectors that represent the image line observations this is not the case. In order to solve this, a constraint is introduced

in the model for each observation vector (Table 2). The advantage of the over-parametrization in combination with a condition is the absence of singularities that can be encountered in other approaches.[13]
Up to this point the observation vectors are specified in the camera system. To be able to combine the observations of different images the camera systems of these images have to be transformed to a common system through their exterior orientation parameters. The transformation of a normal vector $\mathbf{n}$ and a direction vector $\mathbf{d}$ are listed in Table 3.

Table 3: Transformation of observation vectors to a common coordinate system

	direction	*position*	*constraints*
exterior orientation	$\mathbf{R}$ (rotation matrix)	$\mathbf{r}$, $l_{\mathbf{r}}$	$\lvert\mathbf{r}\rvert = 1$
line direction $\mathbf{d}$ (image point)	$\mathbf{Rd}$	$\mathbf{p} = \frac{\mathbf{Rd}\times(\mathbf{r}\times\mathbf{Rd})}{\lvert\mathbf{Rd}\times(\mathbf{r}\times\mathbf{Rd})\rvert}$, $l_{\mathbf{p}} = l_{\mathbf{r}}(\mathbf{r}\cdot\mathbf{p})$	$\lvert\mathbf{d}\rvert = 1$, if $\lvert\mathbf{r}\times\mathbf{Rd}\rvert = 0$ then $l_{\mathbf{p}} = 0$ and e.g. $\mathbf{p} = \frac{(y_{\mathbf{r}}, -x_{\mathbf{r}}, 0)}{\lvert(y_{\mathbf{r}}, -x_{\mathbf{r}}, 0)\rvert}$
plane normal $\mathbf{n}$ (image line)	$\mathbf{Rn}$	$l_{\mathbf{n}} = l_{\mathbf{r}}(\mathbf{r}\cdot\mathbf{Rn})$	$\lvert\mathbf{n}\rvert = 1$

4.2 Parameters or Constraints?

In this section the choice between a mathematical model with object parameters and a model with constraints on the observations is discussed. The model with object parameters is treated first.
Two types of parameters can be distinguished. First there are the parameters that describe the position and orientation of the images, the so-called exterior orientation parameters. Secondly we have the object parameters that describe the geometry of the object. There are many options for the parametrization of the object geometry. Commonly an object geometry is built from points, lines and planes and the parameters of only one type of these object features are used to specify the geometry. The choice for a point, line or plane parametrization is discussed next. In principle parametrization with a mixtures of the three is possible but seems unpractical and will not be considered here.
In case of a parametrization of the object by points or lines constraints have to be applied. In a point parametrization constraints have to ensure that the points that build a face are in one plane. For a line parametrization constraints have to be applied to ensure the intersection of the lines and constraints are needed to ensure the lines are in their planes. The plane parametrization does not need any constraints as (non-parallel) planes always intersect in the lines and points of the topology (as long as not more than three planes intersect in one point otherwise an object constraint is involved, see section 5.2). So the number of parameters and the number of constraints depends on the choice of parametrization. In Table 4 these numbers are listed for two example objects.

Table 4: Number of parameters and constraints of a tetrahedron and a hexahedron

Parametrization by	*tetrahedron # parameters*	*tetrahedron # constraints*	*hexahedron # parameters*	*hexahedron # constraints*
points (3 par.)	12	non	24	6
lines (4 par.)	24	12	48	30
planes (3 par.)	12	non	18	non

Because there is no need for constraints in case of a plane parametrization it seems attractive but there are two major drawbacks. First the plane parametrization can only be applied to describe the geometry of solids. A single or several connected planes in space can not be parametrized with only the parameters of the plane(s) as they are not bounded. The second drawback is the fact that the plane parameters can only be related to the line (and point) observations in the images through the lines in space being the intersections of the planes. This could lead to the conclusion that the line parametrization is to be preferred as e.g. in Ref.13, 19 and 20. But the number of parameters and constraints involved is a disadvantage in this case. The point parametrization is a compromise from the point of view of the number of parameters and constraints and in terms of establishing the relations between observations and parameters.

Independent of the parametrization chosen the mathematical model consists of condition equations with observations and parameters. In addition constraints have to be imposed. This model can be written as follows (after linearization):[18, 22]

$$\mathbf{B}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{A}\mathbf{x}\,,\ \mathbf{C}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{0}\,,\ \mathbf{D}^{\mathrm{T}}\mathbf{x} = \mathbf{0}\,;\ \mathbf{Q}_{\mathbf{y}} \tag{1}$$

with:

$E\{\}$	mathematical expectation
$\mathbf{y}$	vector of observations (image lines and points)
$\mathbf{x}$	vector of parameters (object parameter and exterior orientation parameters)
$\mathbf{A}$, $\mathbf{B}$	design matrices
$\mathbf{C}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{0}$	constraints on the observations (see Table 2)
$\mathbf{D}^{\mathrm{T}}\mathbf{x} = \mathbf{0}$	constraints on the object parameters
$\mathbf{Q}_{\mathbf{y}}$	covariance matrix of the observations

It can be advantageous to regard the constraints on the object parameters as observations as in Ref. 1 and 21 but then the adjusted object parameters do not fully satisfy the constraints. Although the resulting model might be more realistic because of differences between the as-built and the designed geometry, often the user is not interested in these discrepancies.

We will avoid this rather complex model (1) by rewriting the condition equations and converting the constraints on the object parameters to constraints on the observations. In this way the object parameters can be eliminated from the mathematical model but the exterior orientation parameters remain. Without object parameters in the adjustment they have to be computed in a separate step in the so-called object reconstruction (see section 7). In the object reconstruction step adjusted observations (image lines and points) are used to compute parameters of the object points.

To avoid the object parameters, conditions have to be imposed on the observations to ensure that:

- object points are on lines that are related to image points (collinearity condition)
- object lines are in the planes that relate to image lines (image-object coplanarity condition)
- object lines are in their related object planes (object-object coplanarity condition)

These conditions can be called topology constraints because they result from the image-object topology as well as the topology of the object itself. Geometric object constraints like parallelism and perpendicularity of lines or planes is a second type of constraints sometimes called internal constraints.[1, 17] These constraints can be introduced in the same way as the topology constraints. Many of these constraints are discussed in detail in the next section. The (linearized) model that we are looking at now is a model with only condition equations (in (2) $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{x}$ are different from (1)):

$$\mathbf{B}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{A}\mathbf{x}\,;\ \mathbf{Q}_{\mathbf{y}} \tag{2}$$

On the right hand side only the exterior orientation parameters appear in the vector of parameters. This formulation has several advantages. If the exterior orientation parameters are known beforehand, they can be eliminated from the model (2) which then reduces to a model with condition equations with only observations (in (3) $\mathbf{B}$ is different from (2)):

$$\mathbf{B}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{0}\ ;\ \mathbf{Q}_{\mathbf{y}} \tag{3}$$

Another way to simplify the model (2) is to regard the exterior orientation parameters as observations as well. The estimates of the exterior orientation parameters obtained with the method described in the next section are introduced as observations in the model. In this approach the stochastic model needs special attention as correlation between image observations and exterior orientation parameters can not be neglected.

An advantage of the model (3) compared to (1) is the absence of object parameters and therefore there is no need for approximate values of these parameters. A second major advantage of the model (3) is the possibility of a sequential adjustment of this model (see section 6). Constraints can be introduced sequentially in the model (1) but then all the parameters involved have to be estimable. During interactive modeling this is not guaranteed for all the parameters. Sequential adjustment allows a separate evaluation of each condition and is especially important in a semi-automatic measurement system. The misclosure of each condition can be tested as soon as the condition is introduced and the observations involved are available. And maybe even more important, the independency of a newly introduced condition in relation to conditions introduced previously can be verified (see section 6). This eliminates one of the main drawbacks of the condition equation approach, namely the need for independent conditions. A remaining drawback is the effort that has to be put in the formulation of the condition equation i.e. the formulation of the constraints as a function of the observations. This is the subject of section 5.

4.3 Approximate Values

There are quite a few alternatives to obtain approximate values for the exterior orientation parameter of the images. A lot of research has concentrated on the so-called perspective three and four point problem.[14, 15] Often space coordinates of object points are assumed to be available. We want to make use of the rectangularity of object features because rectangular shapes appear frequently in man-made objects like buildings. An efficient and direct solution has been derived for which only a parallelogram (or rectangle) in object space is needed.[16] So only information on parallelism is used and exterior orientation parameters can be derived in an arbitrary object coordinate system. If a specific object coordinate system is to be used a similarity transformation has to be applied. With estimates of the exterior orientation parameters these parameters can be eliminated from (2) by regarding them as observations. Then the model (3) can be used if the covariance matrix of the exterior orientation *observations* is available.

5. FORMULATION OF CONDITION EQUATIONS

The condition equations involve only observations and are the result of two types of geometric constraints. This distinction is used by Weik[17] as well. One type of constraints results from (image-object) topological information such as two lines intersecting in one point or three planes intersecting in one line. This type is called topology constraints. The second type results from constraints between object features such as two planes that have to be perpendicular and is called object constraints. Both types are discussed in this section.

Two problems in the formulation of condition equations can arise. The first one is the possibility of dependency between the various constraints. It will not always be possible to avoid the introduction of dependent constraints during modeling. Constraints depending on (combinations of) previously established constraints have to be eliminated from the mathematical model. This can be done by numerically testing the independency of a candidate constraint. This is part of the sequential adjustment that is the subject of section 6. The second problem is the choice for a particular formulation of a condition equation. There can be many alternatives in the choice of the observations that are used to specify a constraint. The constraint should be formulated in such a way that the condition of the system of normal equations is optimized.

First the topology constraints are presented in section 5.1. The object constraints are discussed in section 5.2.

5.1 Topology Constraints

The topology constraints are constraints that result from the image-object topology. For instance all the lines in the images that are related to one line of the object define planes that have to intersect at the object line. During the interactive modeling process this type of constraints can be set up automatically based on the topology available. Depending on the observations available the geometrical parameters of the object features can be computed. Whether sufficient observations are present can be continuously verified by the system. Topology constraints can be divided into (object) point, line and plane constraints.

Point constraints. An object point can be related to image points and to image lines. The object point constraint for image points has the form (Figure 4):

$$\left[\mathbf{R}^i \mathbf{d}^i, \mathbf{R}^j \mathbf{d}^j, l_{\mathbf{r}}^i \mathbf{r}^i - l_{\mathbf{r}}^j \mathbf{r}^j\right] = 0 \tag{4}$$

with:

$[\ ]$	determinant
$\mathbf{R}^i, \mathbf{r}^i, l_{\mathbf{r}}^i$	exterior orientation parameters of image *i* (Table 3)
$\mathbf{d}^i$	observation vector of the object point in image *i*

This coplanarity condition is used for many other constraints as well. The number of independent constraints per object point of this type is 2 ·(#images for the object point) – 3, with two or more images per point. If there are more than two images per point two independent constraints have to be built for each image except two, using the observations of two other images.

In the sequel the rotation matrix of the exterior orientation will be left out for clarity. Observation vectors are assumed to be rotated into a common coordinate system.

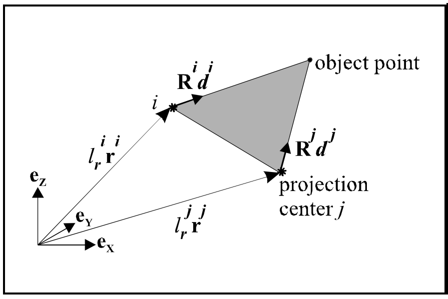

Figure 4: Object point constraint

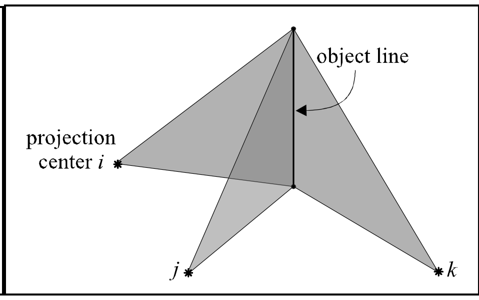

Figure 5: Object line constraint

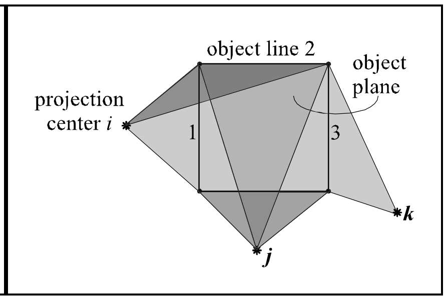

Figure 6: Object plane constraint

The object point constraint for image lines has the same form as the constraint for image points if at least two lines related to an object point are available in each image. Then $\mathbf{d}^i$ is replaced by $\mathbf{n}_1^i \times \mathbf{n}_2^i$ in (4), i.e. the direction of the intersection of the two planes in image *i*. If there are more than two lines in an image related to one object point additional constraints are needed to ensure the intersection of the image lines in one point in the image:

$$\left[\mathbf{n}_1^i, \mathbf{n}_2^i, \mathbf{n}_3^i\right] = 0 \qquad (5)$$

Where $\mathbf{n}_1^i$ is the normal vector to the plane associated with line 1 in image *i*. The number of independent constraints per object point of this type is (#image lines for the point) – 2.

Line constraints. For object lines we have to distinguish between direction and position constraints. The direction constraint is a coplanarity constraint on the normal vectors to the planes associated with image lines and has the following form (Figure 5):

$$\left[\mathbf{n}_1^i, \mathbf{n}_1^j, \mathbf{n}_1^k\right] = 0 \qquad (6)$$

The number of independent constraints per object line of this type is (#images for the object line) – 2.
The position constraints for object lines have the following form:

$$\frac{\left|\mathbf{n}^i \times \mathbf{n}^{ij}\right|}{\left|\mathbf{n}^i \times \mathbf{n}^j\right|^2} = \frac{\left|\mathbf{n}^k \times \mathbf{n}^{jk}\right|}{\left|\mathbf{n}^k \times \mathbf{n}^j\right|^2} \qquad (7)$$

Here $\mathbf{n}^{ij} = (l_\mathbf{r}^i \mathbf{r}^i - l_\mathbf{r}^j \mathbf{r}^j) \times (\mathbf{n}^i \times \mathbf{n}^j)$ is the normal to the plane through the projection centers *i* and *j*. This plane is parallel to the object line. The number of independent constraints per object line of this type is (#images for the object line) – 2.

Plane constraints. The plane constraints ensure that all object lines that border an object plane are parallel to that plane. These object lines are intersecting due to the object point constraints described above and therefore they will not only be parallel to, but in this plane as well. The object plane constraints for direction have the following form (Figure 6):

$$\left[\mathbf{n}_{12}^{ij}, \mathbf{n}_3^i, \mathbf{n}_3^j\right] = 0 \qquad (8)$$

Where $\mathbf{n}_{12}^{ij} = (\mathbf{n}_1^i \times \mathbf{n}_1^j) \times (\mathbf{n}_2^i \times \mathbf{n}_2^j)$ the normal to the object plane constructed from the lines 1 and 2 in images *i* and *j*. Other combinations of images can be used to set up the constraint as well. The number of independent constraints per object plane is (#object lines for the plane) – 2, where only object lines observed in at least two images should be counted.

The need for the plane constraint for position arises for lines in an object plane that are not part of its boundary because then the position of the lines in the plane is not warranted by the point constraint. This is a special case from the geometric modeling point of view. We will not consider this type of constraint here.

5.2 Object Constraints

The topology constraints on the observations presented in the previous section ensure that planes associated with image lines intersect at the points, lines and planes of the object. Object constraints are constraints that hold between object features. We distinguish between constraints on the position of the object features and constraints on the orientation of the features. An overview of the object constraints is given in Table 5.

Table 5: Overview of object constraints

Feature	*point*	*line*	*plane*
point (position)	distance	(shortest) distance	(shortest) distance
line (position)		(shortest) distance	distance (in case of parallelism)
plane (position)			distance (in case of parallelism)
line (orientation)	non	angle	angle
plane (orientation)	non		angle

A separate category of object constraints are the symmetry constraints. These constraints relate three or four object features instead of two. Replacing "distance" by "distance ratio" and "angle" by "angle ratio" in Table 5 results in an overview of this type of constraints. We will not discuss the object symmetry constraints in more detail but present the angle constraint between two lines as an example.

The angle (α) between two object lines can be written as a function of the image line observations:

$$\cos(\alpha) = \frac{(\mathbf{n}_1^i \times \mathbf{n}_1^j)\cdot(\mathbf{n}_2^i \times \mathbf{n}_2^j)}{\left|\mathbf{n}_1^i \times \mathbf{n}_1^j\right|\left|\mathbf{n}_2^i \times \mathbf{n}_2^j\right|} \tag{9}$$

Where $\mathbf{n}_1^i$ is the normal to the plane through the observed line 1 in image i. Both lines have to be observed in at least two images. This constraint can be specified for each pair of object lines. Of course dependency of the constraints could arise. Perpendicularity ($\cos(\alpha) = 0$) and parallelism ($\cos(\alpha) = 1$) are two special cases of this constraint. Parallelism leads to additional coplanarity constraints of the form:

$$[\mathbf{n}_1^i, \mathbf{n}_2^j, \mathbf{n}_3^k] = 0 \tag{10}$$

In this formulation the constraint involves three object lines observed in three different images. In fact the constraint can be specified for any combination of three line observations of object lines that are parallel. This includes line observations in the same image. In that case the direction in space of the object line is found by the intersection of the planes associated with the image line observations. The image lines then intersect in the so-called vanishing point and constraint (10) is identical to (5). The number of independent constraints of this type is (#line observations of parallel object lines) – 2. But these constraints are fully dependent on the constraints (9). Note the dependency on the line constraints (6) involving a single object line.

To derive the constraints of Table 5 in terms of the point and line observation vectors generally leads to a formulation that is more complex than the formulation in terms of object parameters. The complexity of the object constraints in terms of observations is a disadvantage of the mathematical model with only condition equations with observations.

6. SEQUENTIAL ADJUSTMENT

The objective of a sequential adjustment is an adjustment in steps where in each step condition equation(s) and observations can be added to the mathematical model. The condition equation model (3) can be partitioned in two parts in the following way:

$$\begin{pmatrix} \mathbf{B}_1{}^{\mathrm{T}} \\ \mathbf{B}_2{}^{\mathrm{T}} \end{pmatrix} E\{\mathbf{y}\} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \;;\; \mathbf{Q}_{\mathbf{y}} \tag{11}$$

The first part of the model ($\mathbf{B}_1{}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{0}$) represents the condition equations processed previously. The second part ($\mathbf{B}_2{}^{\mathrm{T}} E\{\mathbf{y}\} = \mathbf{0}$) consists of only one condition equation namely the condition that is to be investigated before being added to the system. The solution of the first part of (11) is:[18]

$$\hat{\mathbf{y}}_1 = \mathbf{P}_{\mathbf{B}1}^{\perp}\mathbf{y} \;,\; \mathbf{Q}_{\hat{\mathbf{y}}_1} = \mathbf{P}_{\mathbf{B}1}^{\perp}\mathbf{Q}_{\mathbf{y}} \tag{12}$$

with: $$\mathbf{P}_{\mathbf{B}1}^{\perp} = \mathbf{I} - \mathbf{Q}_{\mathbf{y}}\mathbf{B}_1(\mathbf{B}_1^{\mathrm{T}}\mathbf{Q}_{\mathbf{y}}\mathbf{B}_1)^{-1}\mathbf{B}_1^{\mathrm{T}} \tag{13}$$

Then the new condition equation is added:

$$\mathbf{B}_2^{\mathrm{T}} E\{\hat{\mathbf{y}}_1\} = 0 \ , \ \mathbf{Q}_{\hat{\mathbf{y}}_1} \tag{14}$$

The solution is found in the same way as for the first part of the model but now the adjusted observations are used with their (propagated) covariance matrix $\mathbf{Q}_{\hat{\mathbf{y}}_1}$:

$$\hat{\mathbf{y}} \ = \hat{\mathbf{y}}_{2,1} = \mathbf{P}_{\mathbf{B}2,1}^{\perp}\hat{\mathbf{y}}_1 \ , \ \mathbf{Q}_{\hat{\mathbf{y}}} = \mathbf{Q}_{\hat{\mathbf{y}}_{2,1}} = \mathbf{P}_{\mathbf{B}2,1}^{\perp}\mathbf{Q}_{\hat{\mathbf{y}}_1} \tag{15}$$

with: $$\mathbf{P}_{\mathbf{B}2,1}^{\perp} = \mathbf{I} - \mathbf{Q}_{\hat{\mathbf{y}}_1}\mathbf{B}_2(\mathbf{B}_2^{\mathrm{T}}\mathbf{Q}_{\hat{\mathbf{y}}_1}\mathbf{B}_2)^{-1}\mathbf{B}_2^{\mathrm{T}} \tag{16}$$

Several remarks to this approach have to be made:

- In principle the full covariance matrix of the observations has to be stored and processed. In practice the covariance matrix will retain a certain degree of sparsity because each constraint involves only a small number of observations. With the amount of memory available in nowadays computers storage is not a serious disadvantage of the sequential adjustment.
- There is no matrix inversion needed for a new condition equation as the matrix $(\mathbf{B}_2^{\mathrm{T}}\mathbf{Q}_{\hat{\mathbf{y}}_1}\mathbf{B}_2)$ is identical to a scalar.
- If $(\mathbf{B}_2^{\mathrm{T}}\mathbf{Q}_{\hat{\mathbf{y}}_1}\mathbf{B}_2)$ approaches zero the new condition depends on constraints previously included in the model. In case of dependency the new condition is already satisfied and should (can) not be included in the model.
- For each constraint the misclosure can be computed and statistically tested. In this way possible errors can be detected and corrected during interactive modeling.

7. OBJECT RECONSTRUCTION

The adjusted observations i.e. the image lines and points are input to the object reconstruction process depicted in Figure 7. After adjustment the observations are consistent in the sense that the image points and lines correspond to lines and planes in space that intersect in the points, lines and planes of the object. Furthermore the constraints on the object features like parallelism are satisfied.

Before reconstruction the observations are transformed into the object coordinate system using the exterior orientation parameters of the images. Two types of reconstruction operations are needed. These are the intersection and the construction operation. The first one can be split up in line-line, plane-plane and line-plane intersection. These operations are discussed next. To assess the precision of object features the covariance matrix of the adjusted observations has to be propagated to the covariance matrix of the object parameters.

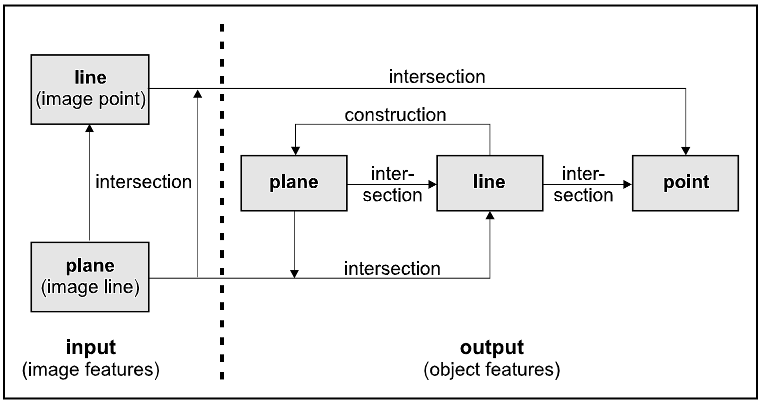

Figure 7: Object reconstruction flow chart

Line-line intersection. The direction to the point that results from the intersection of lines 1 and 2 follows from:

$$\mathbf{x} = \frac{(\mathbf{p}_1 \times \mathbf{d}_1) \times (\mathbf{p}_2 \times \mathbf{d}_2)}{|(\mathbf{p}_1 \times \mathbf{d}_1) \times (\mathbf{p}_2 \times \mathbf{d}_2)|} \tag{17}$$

The length of the vector to the object point results from the projection of the position vector of the lines:

$$l_x = \frac{l_{p1}}{(\mathbf{x} \cdot \mathbf{p}_1)} = \frac{l_{p2}}{(\mathbf{x} \cdot \mathbf{p}_2)} \tag{18}$$

A special case arises if $|(\mathbf{p}_1 \times \mathbf{d}_1) \times (\mathbf{p}_2 \times \mathbf{d}_2)|$ approaches zero in other words if the four vectors defining the two lines are in or near to one plane. Then the line resulting from the intersection of the planes defined by the line position vectors $l_{p1}\mathbf{p}_1$ and $l_{p2}\mathbf{p}_2$ has to be intersected with one of the two original lines. This involves a plane-plane intersection described next.

Plane-plane intersection. The direction of the line that results from the intersection of planes 1 and 2 follows from:

$$\mathbf{d} = \frac{(\mathbf{n}_1 \times \mathbf{n}_2)}{|\mathbf{n}_1 \times \mathbf{n}_2|} \tag{19}$$

If $|\mathbf{n}_1 \times \mathbf{n}_2|$ approaches zero the two planes are (nearly) parallel and an intersection can not be computed. If this is not the case the position vector results from:

$$l_p\mathbf{p} = \frac{l_{n1}(\mathbf{n_2} \times \mathbf{d})}{\mathbf{n}_1 \cdot (\mathbf{n_2} \times \mathbf{d})} + \frac{l_{n2}(\mathbf{n}_1 \times \mathbf{d})}{\mathbf{n}_2 \cdot (\mathbf{n}_1 \times \mathbf{d})} = \frac{l_{n1}(\mathbf{n_2} \times (\mathbf{n}_1 \times \mathbf{n}_2)) - l_{n2}(\mathbf{n}_1 \times (\mathbf{n}_1 \times \mathbf{n}_2))}{|\mathbf{n}_1 \times \mathbf{n}_2|^2} \tag{20}$$

From which l_p and $\mathbf{p}$ can be derived.

Line-plane intersection. The intersection of a line and a plane can be performed by a plane-plane intersection followed by a line-line intersection. The plane-plane intersection uses the original plane and the position vector of the line as a plane definition. In the second step the original line is intersected with the line resulting from the plane-plane intersection. In this way there is no need for an explicit formulation of the line-plane intersection.

Construction of a plane from intersecting lines. The normal vector of the plane defined by intersecting lines 1 and 2 is found from:

$$\mathbf{n} = \frac{(\mathbf{d}_1 \times \mathbf{d}_2)}{|\mathbf{d}_1 \times \mathbf{d}_2|} \tag{21}$$

If $|\mathbf{d}_1 \times \mathbf{d}_2|$ approaches zero the two lines are (nearly) parallel and the plane remains undefined. If this is not the case the distance of the plane to the origin results from the projection of the position vector of one of the lines:

$$l_n = l_{p1}(\mathbf{p}_1 \cdot \mathbf{n}) = l_{p2}(\mathbf{p}_2 \cdot \mathbf{n}) \tag{22}$$

Object constraints can aid in object reconstruction. These constraints are satisfied because they have been applied to the observations (see section 5.2). But parts of the object model might not be computable without them, that is with image observations only. This could occur if redundancy is low (or absent) and many object constraints are included in the model. The latter is true if polyhedral primitives are used for CSG (see section 3). In that case the object reconstruction is split up in two parts. First the shape parameters of the primitives and their position and orientation are determined from the observations through the intersection and construction operations described above. The (relative) positions and orientations of the primitives satisfy the constraints imposed on them. In the second part of the reconstruction the CSG-tree grown in the modeling process is evaluated and the primitives are combined using CSG-operations resulting in the final model of the object or building.

8. CONCLUSIONS AND FUTURE WORK

A novel formulation of a line-photogrammetric mathematical model was presented. The proposed model is based on condition equations with observations only. Image pose parameters and object parameters are removed from the model. Estimates of the image pose parameters are determined using only information on parallelism in object space. Knowing the pose parameters they can be eliminated from the mathematical model. Constraints on the object parameters are transformed to constraints on the observations. The advantage of this approach is the absence of parameters in the mathematical model and therefore there is no need for approximate values of these parameters. Furthermore the model facilitates sequential adjustment so possible errors can be detected during interactive modeling. Disadvantages are the additional processing step i.e. the separate object reconstruction and the complexity of the formulation of the geometric object constraints as a function of the observations. This type of constraints is especially frequent in CSG-primitives as these shapes are defined by only a few parameters. In the approach presented here primitives have to be represented by a combination of topology and constraints. Modeling with CSG will improve efficiency considerably as with each primitive the topology and constraints of many object features are provided. The research will continue with a final choice of an integrated data structure for image observations and object parameters. The choice of a mathematical model, the CA(A)D environment to be used and the solid modeling kernel will be the starting-point for the implementation. Then the research will be directed towards a knowledge-based approach to image analysis.

REFERENCES

1. A. Streilein, U. Hirschberg, "Integration of digital photogrammetry and CAAD: constraint-based modelling and semi-automatic measurement", *CAAD Futures '95 International Conference*, Singapore, 1995
2. F. Lang, W. Förstner, "Surface reconstruction of man-made objects using polymorphic mid-level features and generic scene knowledge", *International Archives of Photogrammetry and Remote Sensing*, Vol.31, Part B3, pp.415-420, Vienna, 1996
3. S. Mason, "3D Building Reconstruction using Composites of Surface Primitives: Concept", *International Archives of Photogrammetry and Remote Sensing*, Vol.31, Part B3, pp.517-522, Vienna, 1996
4. Y. Hsieh, "SiteCity: A semi-automated site modelling system", *IEEE conference on Computer Vision and Pattern Recognition*, pp.499-506, San Francisco, 1996
5. W. Förstner, "A Framework for Low Level Feature Extraction", *Computer Vision - ECCV '94*, Vol. II, pp.383-394, Ed.: J. O. Eklundh, LNCS 802, Springer, 1994
6. J.B. Burns, A.R. Hanson, E.M. Riseman, "Extracting Straight Lines", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No.4, pp.425-455, 1986
7. F. Lang, W. Förstner, "3D-city modeling with a digital one-eye stereo system", *International Archives of Photogrammetry and Remote Sensing*, Vol.31, Part B4, pp.261-265, Vienna, 1996
8. M. Sinning-Meister, A. Gruen, H. Dan "3D City models for CAAD-supported analysis and design of urban areas", *ISPRS Journal*, nr.51, pp.196-208, 1996
9. P.E. Debevec, C.J. Taylor, J. Malik, "Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach", *Computer Graphics proceedings, Annual Conference Series*, pp.11-20, New Orleans, 1996
10. M.E. Mortenson, *Geometric modeling*, John Wiley & Sons, Inc., ISBN 0-471-12957-7, New York, 1997
11. M. Molenaar, "A topology for 3D vector maps", *ITC Journal* nr.1, pp.25-33, 1992
12. K. Kanatani, *Statistical optimization for geometric computation: theory and practice*, Machine Intelligence and Pattern Recognition Series, Vol.18, Elsevier Science B.V., Amsterdam, 1996
13. P. Patias, E. Petsa, A. Streilein, *Digital Line Photogrammetry*, IGP Bericht nr. 252, Eidg. Technische Hochschule, ISBN 3-906513-73-4, Zürich, 1995
14. R. Horaud, B. Conio, O. Leboullieux, "An analytic solution for the perspective 4-point problem", *Computer Vision, Graphics, and Image Processing* 47, pp.33-44, 1989
15. R. M. Haralick, C.-N. Lee, Ottenberg, M. Noelle, "Review and analysis of solutions of the three point perspective pose estimation problem", *Int. Journal of Computer Vision*, Vol.13, No.3, pp.331-356, 1994
16. F.A. van den Heuvel, "Exterior Orientation using Coplanar Parallel Lines", *10th Scandinavian Conference on Image Analysis*, (in press), Lappeenranta (Finland), 1997
17. S. Weik, O. Grau, "Recovering 3-D object geometry using a generic constraint description", *International Archives of Photogrammetry and Remote Sensing*, Vol.31, Part B3, pp.593-598, Vienna, 1996
18. P.J.G. Teunissen, *Adjustment Theory*, Lecture notes Faculty of Geodetic Engineering, Delft University of Technology, Delft, 1994
19. H. Zielinski, *Object reconstruction with digital line photogrammetry*, Doctoral thesis Department of Geodesy and Photogrammetry, Royal Institute of Technology, Stockholm, 1993
20. E. Guelch, "Line photogrammetry: a tool for precise localization of 3D points and lines in automated object reconstruction", *Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision II*, SPIE Vol. 2486, pp.2-12, Orlando, 1995
21. J.C. McGlone, "Bundle adjustment with object space geometric constraints for site modeling", *Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision II*, SPIE Vol. 2486, pp.25-36, Orlando, 1995
22. E.M. Mikhail, *Observations and Least Squares*, IEP-A Dun-Donnelley Publisher, New York, 1976