

A GENERIC PROCEDURE FOR INTERACTIVE 3D MODEL REGISTRATION ON IMAGES

Philippe EVEN

LORIA / IUT de St-Dié,
11, Rue de l'Université, F-88100 Saint Dié des Vosges, France,
Philippe.Even@loria.fr

KEY WORDS: Three-dimensional, Model, Image, Registration, User, Interface.

ABSTRACT

This paper presents a generic and user-friendly procedure for interactive 3-D model registration on images. Such tasks are numerous in a CAD-based model acquisition process through interactive means. Moreover object alignment can also be used to quickly provide an initial match when registering a whole model on actual data. This task could be performed by setting a minimal set of 2D-3D point matches. Our goal is to set up a more user-friendly solution, which minimizes the amount of manual work. We have implemented a two-step procedure relying on the selection in the image of some structured object features. First a visual feature interpretation method is used. It provides a solution that fixes at least two rotation parameters. Remaining degrees-of-freedom are then easily handled through interactive modes. Several interpretation methods are proposed to cope with various contexts according to the available features. Achieved accuracy is *a posteriori* estimated to help the operator decide on the registration reliability.

1 INTRODUCTION

The theme of our research group at St-Dié deals with task-oriented 3-D modelling of existing environments (Even, 2004). Compared to other teams working in the computer vision domain, our goal focuses on the acquisition of 3-D models that are well adapted to given tasks. Typical requirements include acquisition time, geometrical reliability or visual realism, but also relevant structuration and semantical contents. The operator's knowledge of the model destination and his ability to select appropriate details in the sensed data are necessary and manual interventions are thus required within the modelling process. Therefore we put emphasis on interactive methods with efficient computer assistances that are based on geometrical reasoning, image processing or expertise integration.

We chose a top-down approach (from 3D to 2D) where solid primitives are selected and assembled together directly on the images in order to ensure the conformity to reality (Even, 2001). It provides a more natural way to build the 3-D model than a classical bottom-up approach (from 2D to 3D) where homologous image features are first extracted and then processed to basically build a surface model. Its high flexibility to cope with hard environment constraints and erroneous hypotheses was recently demonstrated in the scope of a radioactive cell modelling work (Leroux et al., 2002).

This interactive modelling method includes many objects registration tasks to make their projection match relevant image features. They are mainly intended to put objects at their correct place in 3-D space, assuming that the background image is oriented. But inversely, they are also used to orient new images using already modelled objects. The object degrees of freedom (3 translations + 3 rotations) are successively controlled with the mouse. Most often this work proves uneasy, mainly because it is difficult to determine the rotation axis. In structured environment, some

geometrical constraints can often be used to back-project image features such as points, lines, or elliptical contours. They deal with parallelism, orthogonality, axial symmetry, etc. Each one may require a dedicated interface, and so globally hamper the system appropriation by the user.

Therefore we implemented a generic two-steps procedure. Based on the available constraint, a direct solution is first computed in order to fix the rotation axis. The remaining degrees of freedom are then interactively determined using the mouse. An estimation of the solution consistency is finally displayed to the user so that he could decide of the registration reliability.

2 ANALYTICAL SOLUTIONS TO THE REGISTRATION PROBLEM

Our top-down approach aims at limiting the number of visual feature extraction tasks. Many authors already addressed the problem of registration from a restricted set of points or lines. In the RANSAC paradigm (Fischler and Bolles, 1981), a small set of n control points is used to provide a first solution, further enhanced with consistent additional points. It was shown that a unique solution can be computed from four coplanar points. Analytical solutions were later proposed to solve this *perspective-4-points* problem with different angular configurations between the points (Horaud et al., 1989).

The pose determination problem can also be solved from lines. It was shown that the projection of three lines with known relative orientations leads to an eighth-degree equation (Dhome et al., 1989). Actually two vertical lines and a horizontal line are sufficient to directly determine the rotation matrix (Shum et al., 1998). Moreover the case of lines belonging to planes with known configurations was also studied (Chen, 1991); for coplanar configurations where the lines belong to the same plane, orthogonal configurations where the planes are mutually perpendicular, and pa-

parallel configurations where two planes are parallel, found closed-form solutions are the intersections of three hypercones in 4-space.

Recent works on the determination of a circle pose from its projection in a single view led to algebraic (Kanatani and Liu, 1993) or geometric (Chen and Huang, 1999) solutions. A closed-form solution was also proposed for the reconstruction of conics from their projections in a couple of views with known relative orientation (Quan, 1996).

3 THE TWO-STEPS PROCEDURE

3.1 Principle and implementation

The main drawback of a manual registration comes from the determination of the rotation axis. Therefore we implemented a two-steps procedure which sets up a middle course between manual registration and features extraction tasks. First a small set of features is extracted according to the scene and images geometrical configurations in order to fix at least two of the three rotation parameters. The remaining degrees of freedom are then controlled interactively with the mouse.

A set of five configurations is implemented. It is large enough to cope with most structured scenes already encountered, and small enough for a fast appropriation of the user interface. Three of them require two calibrated views, the rigid transformation between both views being known. The two remaining ones use a single view and may also be used to determine the exterior orientation. These configurations are:

- *Stereo line*: selection of one visible line in two views.
- *Shape of revolution*: selection of the occlusion contours of a shape of revolution in two views.
- *Coplanar features*: selection of coplanar points or lines in two views.
- *Parallel lines*: selection of the projection of two parallel lines in a single view.
- *Orthogonal lines*: selection of the projection of three orthogonal lines.

The operating mode is sequenced as follows:

⇒ *Preliminary tasks* : The user first selects the object \mathcal{O} to be registered, a reference frame $\mathcal{R}_o(P_o, \vec{X}_o, \vec{Y}_o, \vec{Z}_o)$ on that object for the interactive manipulation tasks, and the relevant configuration. Default actions move the object on a fixed image. It is left possible to change this behaviour to move the camera position wrt a fixed object and thus determine the camera exterior orientation.

⇒ *First step* : Visual features are selected to compute a solution $\mathcal{R}_s(P_s, \vec{X}_s, \vec{Y}_s, \vec{Z}_s)$ which is used to align \mathcal{R}_o . It is arranged so that \vec{Z}_s corresponds to the fixed rotation axis, P_s is defined by the first selected image point, and \vec{X}_s turns to the viewer. Some configurations admit several solutions. It was preferred to let the user switch afterwards between these solutions, rather than asking him to select additional features to remove possible ambiguities. The closest solution to \mathcal{R}_o is first proposed. Its axes are then quickly

re-arranged if needed.

⇒ *Second step* : Interactive actions are successively performed until a correct registration is achieved.

3.2 Image features 3-D interpretation

Let P be an image point with coordinates (u, v) wrt the image reference system. If the camera interior orientation is known, we can find its coordinates (X_C, Y_C, Z_C) wrt the camera reference system, assuming a unit distance between the image plane I and the optical centre C . Its coordinates (X, Y, Z) wrt the world reference system can then be computed from the exterior orientation matrix. The optical ray \mathcal{L} defined by C and \overline{CP} is the *interpretation line* of P . It is the locus of all the possible interpretations M of P in the 3-D space.

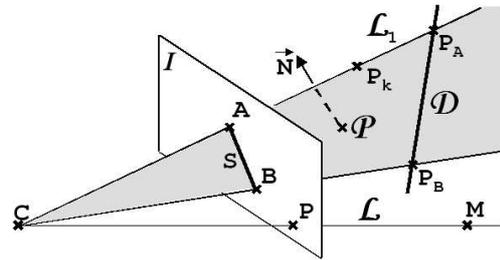


Figure 1: Point and segment geometry.

Now let S be a segment joining image points A and B . It may be the projection of a 3-D line \mathcal{D} to be retrieved. All the possible interpretations of S in the 3-D space lie within a plane \mathcal{P} defined by C and normal vector $\vec{N} = \overline{CA} \wedge \overline{CB}$. \mathcal{P} is the *interpretation plane* of S .

Implemented configurations are detailed in the following. For the ones requiring two views, a prime is used to mark corresponding features on the second view. \mathcal{L}_1 refers to the interpretation line of the first selected point or segment end, and P_k is a point of \mathcal{L}_1 at arbitrary distance k from C .

3.3 Stereo line configuration

This configuration relies on the specification of a visible edge in two images to retrieve the 3-D line \mathcal{D} . Two rotation and two translation parameters are then fixed to constrain the interactive registration.

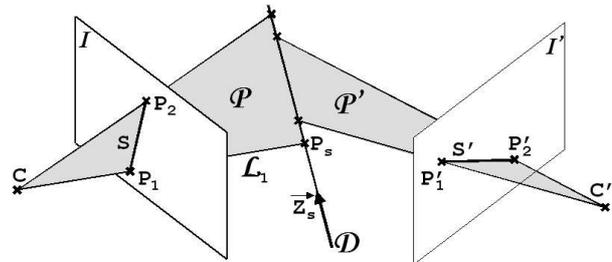


Figure 2: Stereo line geometry.

Let S and S' be the selected segments, and \mathcal{P} and \mathcal{P}' their interpretation planes with normal vectors \vec{N} and \vec{N}' . A

solution can be found, provided that these vectors are not collinear. \mathcal{D} is the intersection of \mathcal{P} and \mathcal{P}' . The computed solution is:

$$\mathcal{R}_s : \begin{cases} P_s = \mathcal{L}_1 \cap \mathcal{P}' \\ \vec{Z}_s = \text{norm}(\vec{N} \wedge \vec{N}') \\ \vec{Y}_s = \text{norm}(\vec{CP}_s \wedge \vec{Z}_s) \end{cases}$$

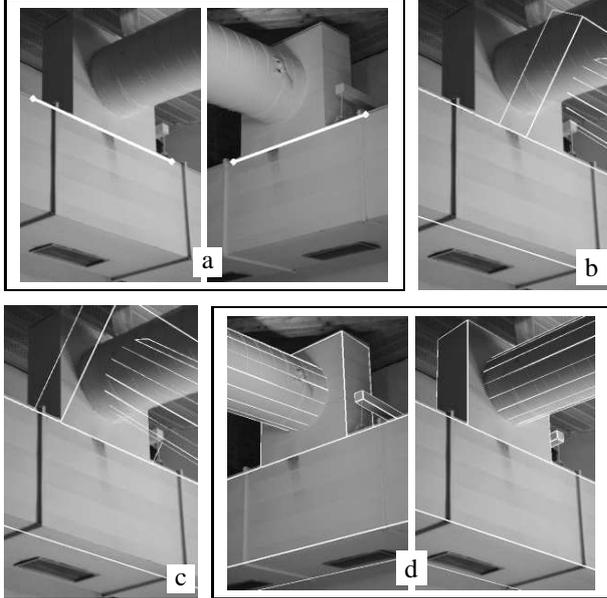


Figure 3: Stereo line example.

Figure 3 example displays the selected features (a) and the matched object on the computed solution (b). Then successive interactive actions are:

- 1) translation along \mathcal{D} (c),
- 2) rotation around \mathcal{D} (d).

Both images may display a distinct part of the 3-D edge. This could be useful to limit the number of views when modelling large objects as no overlap is required.

3.4 Shape of revolution configuration

This solution estimates the axis \mathcal{A} of a shape of revolution from the specification of its contours in two views. Let S_1 and S_2 (resp. S'_1 and S'_2) be the selected segments, and \mathcal{P}_1 and \mathcal{P}_2 (resp. \mathcal{P}'_1 and \mathcal{P}'_2) their interpretation planes with normal vectors \vec{V}_1 and \vec{V}_2 (resp. \vec{V}'_1 and \vec{V}'_2). Axis \mathcal{A} is the intersection of mediator planes \mathcal{P}_M and \mathcal{P}'_M , where \mathcal{P}_M is given by point C and vector $\vec{N}_M = \vec{N}_1 + \vec{N}_2$.

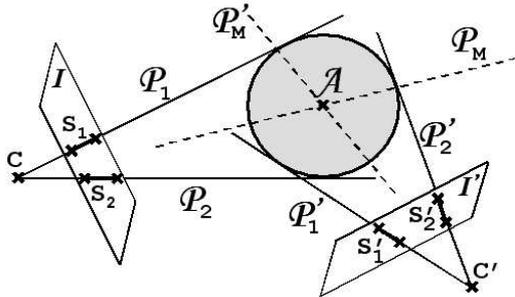


Figure 4: Shape of revolution geometry.

If \vec{N}_M and \vec{N}'_M are not collinear, the solution is:

$$\mathcal{R}_s : \begin{cases} \vec{Z}_s = \text{norm}(\vec{N}_M \wedge \vec{N}'_M) \\ P_s = \mathcal{L}_1 \perp \mathcal{A} \\ \vec{Y}_s = \text{norm}(\vec{CP}_s \wedge \vec{Z}_s) \end{cases}$$

where $a \perp b$ is the orthogonal projection of a on b .

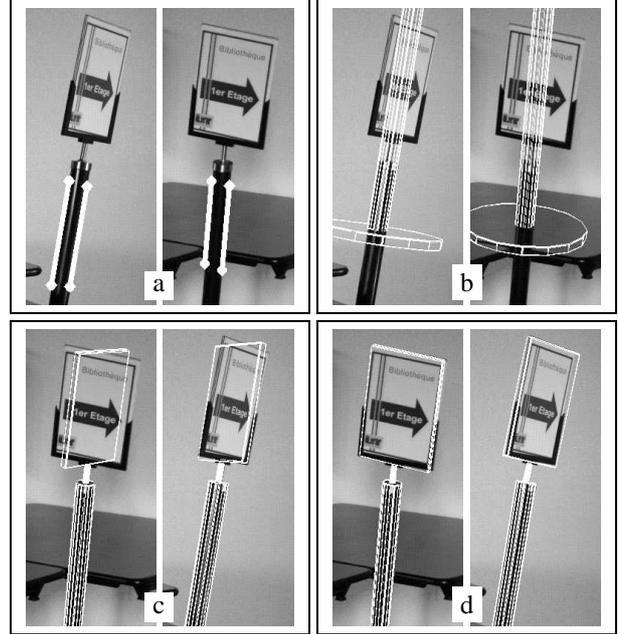


Figure 5: Shape of revolution example.

Figure 5 example displays the selected features (a) and the matched object on the computed solution (b). Then successive interactive actions are:

- 1) translation along \mathcal{A} (c),
- 2) rotation around \mathcal{A} (d).

It is interesting to note that this solution may handle any object that contains in particular a cylindrical or conical part.

3.5 Coplanar features configuration

This solution computes a 3-D plane \mathcal{T} from the selection of a set of coplanar features in two images. It fixes two rotation and one translation parameters. Proposed sets are:

- three points P_i (resp. P'_i), $i \in [1, 3]$; each couple of interpretation lines \mathcal{L}_i and \mathcal{L}'_i theoretically intersects in point M_i . The triplet (M_1, M_2, M_3) defines \mathcal{T} . Actually the interpretation lines do not intersect because of numerical errors in calibration, numerization or manual selection tasks. Therefore each M_i is computed as a rough but sufficient estimation as the middle point between \mathcal{L}_i and \mathcal{L}'_i . The plane \mathcal{T} can be computed if the three points are not aligned.
- one line segment S and one point P (resp. S' and P'); the segments interpretation planes intersect in line \mathcal{D} , and a point M is obtained by triangulation of P and P' (figure 6). If M does not belong to \mathcal{D} , \mathcal{T} is defined by \mathcal{D} and M .

- two line segments S_1 and S_2 (resp. S'_1 and S'_2); The intersection of both interpretation planes \mathcal{P}_i and \mathcal{P}'_i defines a line \mathcal{D}_i that should lie within \mathcal{T} . Because of numerical errors, \mathcal{D}_1 and \mathcal{D}_2 are not coplanar. Here again, a mean-squares estimation of \mathcal{T} is computed from the selected segments.

In the computed solution \mathcal{R}_s , vector \vec{Z}_s is the normal vector of \mathcal{T} , $\mathcal{P}_s = \mathcal{L}_1 \cap \mathcal{T}$, and $\vec{Y}_s = \text{norm}(\overline{CP}_s \wedge \vec{Z}_s)$.

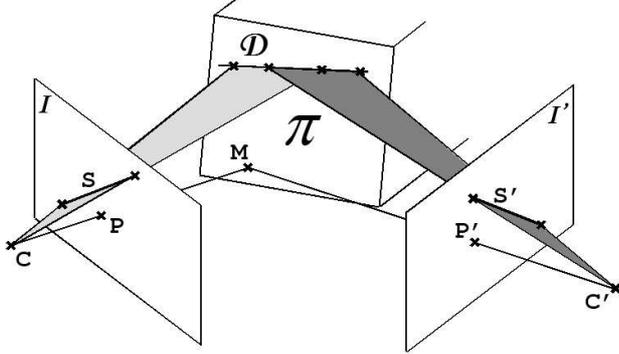


Figure 6: Coplanar features geometry.

The selected features may be some surface details that do not necessarily belong to the 3-D model. The operator is not required to explicitly indicate which set of features he will select. This can easily be guessed afterwards. In the same order of idea, the features are matched automatically. For the three-points set, all the possible combinations are tested. The distance between the interpretation lines of each couple is accumulated and the lightest combination is selected. For the two-segments set, an equivalent test is performed to select the combination that provides the nearest lines to coplanarity.

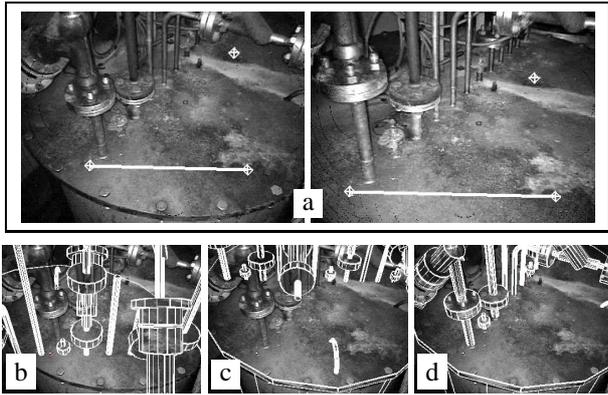


Figure 7: Coplanar features example.

Figure 7 example displays the selected features (a) and the matched object on the computed solution (b). Then successive interactive actions are:

- 1) translation within \mathcal{T} (c),
- 2) rotation around \mathcal{Z}_o (d).

3.6 Parallel lines configuration

This configuration relies on the selection of the projection in a single image of two lines assumed to be parallel in the

3-D space. It provides a direction which is used to facilitate the registration task. Only two rotation parameters are fixed.

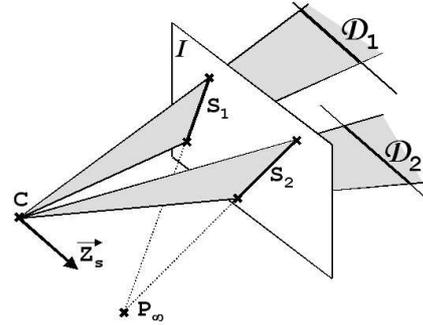


Figure 8: Parallel lines geometry.

The selected segments S_1 and S_2 define two lines which intersect at point P_∞ . In projective geometry, this point P_∞ is known as a *vanishing point* where all lines with a common direction meet. This direction is given by vector \overline{CP}_∞ . The computed solution is :

$$\mathcal{R}_s : \begin{cases} \vec{Z}_s = \text{norm}(\overline{CP}_\infty) \\ P_s = P_k \\ \vec{Y}_s = \text{norm}(\overline{CP}_s \wedge \vec{Z}_s) \end{cases}$$

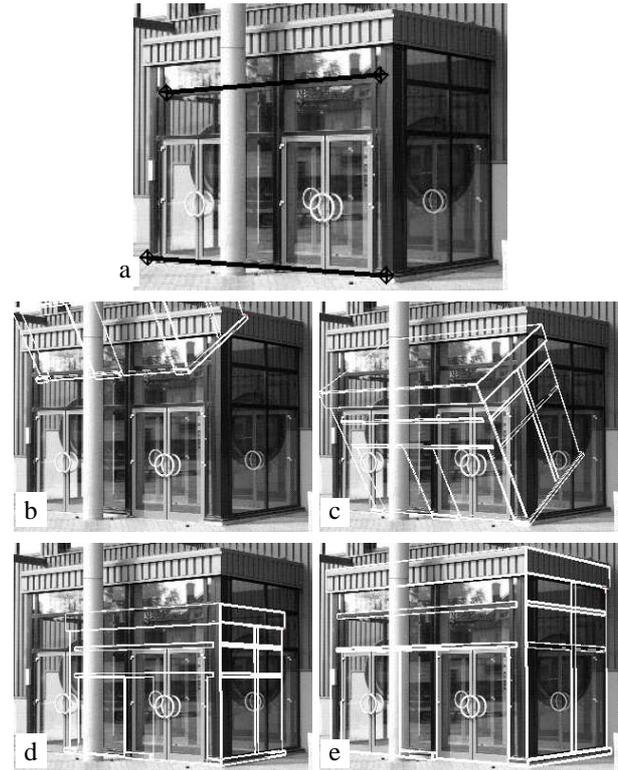


Figure 9: Parallel lines example.

Figure 9 example displays the selected features (a) and the matched object on the computed solution (b). Then successive interactive actions are:

- 1) parallel translation to the image plane I (c),
- 2) rotation around \mathcal{Z}_o (d),
- 3) translation along the interpretation line of P_o (e).

3.7 Orthogonal lines configuration

This solution requires the specification of the projection of three edges assumed to be orthogonal in the 3-D space. It fixes the three object orientation parameters, letting undetermined the translation.

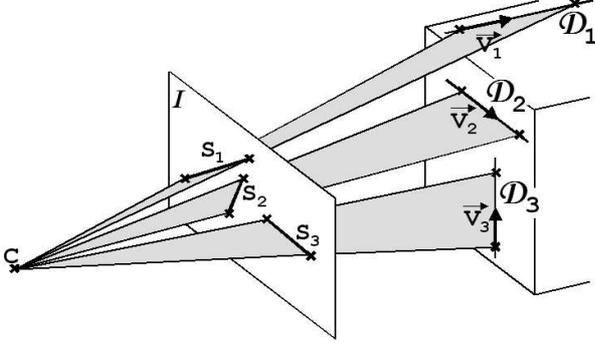


Figure 10: Orthogonal lines geometry.

Let $S_1(A_1B_1)$, $S_2(A_2B_2)$, $S_3(A_3B_3)$ be the selected segments. The director vector \vec{V}_i of the 3-D edge D_i that projects in S_i can be expressed as a linear combination of $\vec{a}_i = \vec{CA}_i$ and $\vec{b}_i = \vec{CB}_i$. As its length is not constrained, a scalar k_i exists so that:

$$\vec{V}_i = k_i \cdot \vec{b}_i - \vec{a}_i, \quad i \in [1, 3]$$

The orthogonality of the 3-D edges is given by:

$$\vec{V}_i \cdot \vec{V}_j = 0, \quad i, j \in [1, 3], i \neq j$$

We must then solve the following system:

$$\begin{cases} k_1 k_2 \cdot \vec{b}_1 \cdot \vec{b}_2 - k_1 \cdot \vec{b}_1 \cdot \vec{a}_2 - k_2 \cdot \vec{a}_1 \cdot \vec{b}_2 + \vec{a}_1 \cdot \vec{a}_2 = 0 \\ k_1 k_3 \cdot \vec{b}_1 \cdot \vec{b}_3 - k_1 \cdot \vec{b}_1 \cdot \vec{a}_3 - k_3 \cdot \vec{a}_1 \cdot \vec{b}_3 + \vec{a}_1 \cdot \vec{a}_3 = 0 \\ k_2 k_3 \cdot \vec{b}_2 \cdot \vec{b}_3 - k_2 \cdot \vec{b}_2 \cdot \vec{a}_3 - k_3 \cdot \vec{a}_2 \cdot \vec{b}_3 + \vec{a}_2 \cdot \vec{a}_3 = 0 \end{cases}$$

k_2 is extracted from the first equation and k_3 from the second one. Substitution in the third equation provides a quadratic equation in k_1 . This system may admit two real solutions that can be used to derive k_2 and k_3 , and then the three vectors \vec{V}_i . Each solution corresponds to the direct or the inverse interpretation of the built reference frame. If the equation has no real root, the provided segments configuration cannot be associated to a set of orthogonal segments. This may come from numerical errors when the viewpoint configuration is close to some limit condition. Our implementation displays the closest solution to the configuration where the built reference frame faces towards the viewer with a vertical z axis. \mathcal{R}_s is built over \vec{V}_1 , \vec{V}_2 and \vec{V}_3 , with $P_s = P_k$.

Figure 11 example displays the selected features (a) and the matched object on the computed solution (b). Then successive interactive actions are:

- 1) parallel translation to the image plane I (c),
- 2) translation along P_o interpretation line (d).

This solution is quite general and in particular, each segment may be indifferently selected in a separate view. Actually our experience showed that finding a view with a

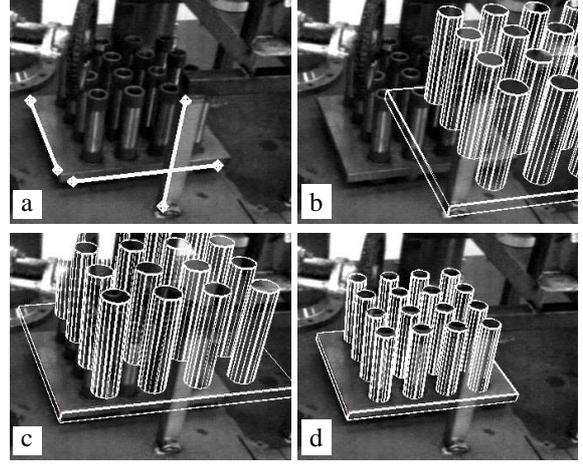


Figure 11: Orthogonal lines example.

long edge in each orthogonal direction can be a really hard constraint. Two long orthogonal edges may be easily displayed, but most often the third one is short and may deteriorate the solution accuracy.

4 ADDITIONAL ASSISTANCES

4.1 Semi-automatic segment extraction

All these solutions require the extraction of image segments. This task repetition could become tedious and soon impede the system performance. Therefore we have implemented a semi-automatic segment extraction function in which the operator firstly draws a rough initial segment close to the relevant edge, and lets it be attracted towards the best contour found in the surrounding area (Even and Malavaud, 2000). The way it was integrated is particularly well suited to cope with the many possible failures of the automatic attraction function, so that this assistance actually contributes to speed up the modelling work.

4.2 Estimation of the solution consistency

A selection of the fewest image features possible makes our technique very sensitive to the many error sources that are linked to sensor definition, interior and exterior orientation inaccuracy, selected image features quality (sharpness, contrast) and care of the interactive delineation tasks. Moreover computer vision algorithms are viewpoint dependent and these primary error sources are amplified by geometrical conditions such as the features spatial distribution or the viewpoints relative position. The operator has not always a good perception of the actual geometrical configuration. This drawback is amplified by the expertise level required. In a given situation, estimating if the features distribution and viewing conditions are favourable or not requires a fine knowledge of the implemented algorithm.

Some estimation of the geometrical configuration quality must be computed and provided to the operator to help him decide on the registration reliability according to the task

requirements. Unfortunately, this cannot be done *a priori*. But as soon as one of the solutions is computed, we have all the necessary information to display this quality indication. According to the kind of features and the number of views that were used, one of the following measurements is computed.

- For edges extracted from a selected segment in two views, the angle between the two interpretation planes is graphically displayed.
- For triangulated points, the distance between the two interpretation lines is provided. It is expressed both in pixel unit to reflect possible image or pairing errors, and in world reference unit to let the user appreciate the geometrical configuration.
- For solutions based on a single view, the maximum deviation caused by a one-pixel perturbation applied to each segment end is computed.

5 CONCLUSION

Manual registration of 3-D models on images remains a difficult task. We have implemented a generic procedure based on the 3-D interpretation of a restricted set of visual features in order to fix two rotation parameters at least. The registration work can then be very easily completed through interactive means. Different possible geometric configurations are interfaced in a quite homogeneous way, so that this procedure provides the required geometrical assistance without calling in question the high flexibility of our interactive approach to the 3-D modelling work.

It was tested on many applications in structured environments (industry, urban sites, interior scenes), with images or scenes with different levels of quality (low contrasts, low illumination, highly specular surfaces), and with variable knowledge on sensor calibration or scene geometry. In all these situations, the small set of implemented solutions could be used to bring off the modelling. Unfortunately the ergonomics evaluations campaign that was planned could not be held, and up to now we have only feedbacks from some experienced users. We believe that the user interface can be mastered in a very short period. But we lack of information about the required time to acquire enough experience to be able to select the relevant configuration and image features. For instance a typical error is to compute the vanishing point of lines that are nearly parallel to the image plane. The numerical instability of the provided solution causes inconsistencies that are well detected by the quality estimation calculation, and that soon affect the modelling if the user ignores these indications. But lot of time may be lost if it happens too often.

This registration procedure is integrated into the new interactive 3-D modelling software which is still under process in our laboratory. We plan to complete it with a sixth solution based on the interpretation of elliptical contours. This could help in the many situations where circular shapes are available in the displayed environment.

REFERENCES

- Chen, H. H., 1991. Pose determination from line-to-plane correspondences: existence condition and closed-form solutions. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 13(6), pp. 530–541.
- Chen, Z. and Huang, J.-B., 1999. A vision-based method for the circle pose determination with a direct geometric interpretation. *IEEE Trans. on Robotics and Automation* 15(6), pp. 1135–1140.
- Dhome, M., Richetin, M., Lapreste, J. and Rives, G., 1989. Determination of the attitude of 3D objects from a single perspective view. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 11(12), pp. 1265–1278.
- Even, P., 2001. A top-down approach to interactive 3-D modelling of architectural sites. In: *XVIIIth International Symposium of CIPA, Potsdam, Germany*, pp. 585–592.
- Even, P., 2004. Task-oriented 3D reconstruction: relevance, actuality and perspectives. In: *Proc. of 3D Modelling 2004, Paris, France*.
- Even, P. and Malavaud, A., 2000. Semi-automated edge segment specification for an interactive modelling system of robot environments. In: *XIXth ISPRS Congress: Geoinformation for All, Amsterdam, NL*, pp. 222–229.
- Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* 24(6), pp. 381–395.
- Horaud, R., Conio, B., Leboulloux, O. and Lacolle, B., 1989. An analytic solution for the perspective 4-point problem. *Computer Vision, Graphics, and Image Processing* 47(1), pp. 33–44.
- Kanatani, K. and Liu, W., 1993. 3-D interpretation of conics and orthogonality. *Computer Vision, Graphics, and Image Processing* 58(3), pp. 286–301.
- Leroux, C., Even, P., Lottin, A., Gelin, R., Idasiak, J.-M., Boissonneau, J.-F. and Jeanjacques, M., 2002. Flexible 3-D modelling of hostile environments with constrained viewpoint and multiple unknowns. In: *ISPRS Comm. V Symposium: Close-Range Imaging, Long-Range Vision, Corfu, Greece*, pp. 175–180.
- Quan, L., 1996. Conic reconstruction and correspondence from two views. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18(2), pp. 151–160.
- Shum, H.-Y., Han, M. and Szeliski, R., 1998. Interactive construction of 3D from panoramic mosaics. In: *IEEE Conf. on Computer Vision and Pattern Recognition, Santa Barbara, CA*, pp. 427–433.