AUTOMATIC INITIALIZATION FOR THE REGISTRATION OF GIS AND VIDEO DATA

T. Colleu, G.Sourimant, L.Morin

IRISA/INRIA/University of Rennes 1
Campus Universitaire de Beaulieu
Avenue du General Leclerc, 35042 RENNES Cedex - France

ABSTRACT

The context of this study is 3D city modeling. The goal is to automatically compute the initial registration of a GIS model and a video sequence. The proposed method is divided into two parts. First, a coarse registration is obtained using GPS data and the theory of epipolar geometry. Then, a simultaneous pose and correspondence determination is done thanks to RANSAC algorithm applied on line features. Experiments on real sequences show that a correct registration is achieved.

Index Terms— virtual reality, augmented reality, model/image registration, urban areas, Global Positioning System, line detection, RANSAC.

1. INTRODUCTION

3D city modeling has several applications in virtual and augmented reality such as virtual visits, geo-positioning and urbanism. The success of google earth shows that adding photometric information on a 2D map provides a great benefit for the user. Similarly, when a 3D city model is available, synthetic textures can be used but the photo-realism is generally poor. Therefore, it is interesting to map real textures onto geometric information of a 3D city model.

Our solution is based on this framework: we use a Geographical Information System (GIS) that contains the footprint and elevation of each building, and we want to enhance this model with real textures extracted from a video sequence. During the acquisition, the video has been synchronized with the GPS data in order to know the approximative position of the camera in the geo-referenced coordinate frame.

Before the mapping process, the GIS and video information must be registered: for each image of the sequence, we must determine the position and orientation of the camera in the geo-referenced coordinate frame, such that the perspective projection of the GIS in the camera frame is aligned with the buildings in the image.

Registration initialization is the registration of the first video frame. It is more difficult than on successive frames since no previous estimation is available. It comprises two coupled issues: 3D/2D feature matching and pose estimation, each easy to solve only if the other has been solved first. One solution consists in eliminating one problem either with a manual intervention or with a more sophisticated equipment. Hence, in [1] the user indicates manually the correspondences. In [2] and [3], the pose is directly measured with the navigation equipment (GPS + inertial platform). Other solutions are proposed based on more accurate models such as textured model [3] or range data [4].

Our 3D model is very rough. In this case two approaches have been proposed which assume that a coarse pose is already available. The first one uses the RANSAC algorithm [5], it is effective only if the correspondence set is small, with few outliers. The second method is based on the minimization of a cost function [6]. Here, divergence can appear due to the non-linearity of the cost function.

In this paper, we perform the initial registration in two steps. First, we compute a coarse estimation of the pose so that the projected features from the GIS are relatively close to the features in the image. To do so, we identify the relative translation of the camera with the GPS translation. Second, we simultaneously estimate the pose and a correspondence set using the RANSAC algorithm. The reduction of the possible features in the image is done thanks to computer vision methods.

These two steps are presented in the sections 2 and 3 respectively. Section 4 gives the registration results obtained with real data.

2. COARSE ESTIMATION OF THE POSE

2.1. Principle

Let $P_i = K[R_i|t_i]$ be the projection matrix of the camera at time i, with K the matrix of intrinsic parameters. We assume that K is known thanks to a preliminary calibration step. Let $[R_i|t_i]$ the matrix of extrinsic parameters defining the orientation and position of the camera in the reference frame. R_i is a rotation matrix and t_i is a translation vector.

We want to estimate the initial pose of the camera, i.e. R_1 and t_1. Considering that the GPS data provide an approximate value $\tilde{t_1}$ for t_1, we still have to find the value of R_1.

Our approach consists in matching the translation that can be extracted from the video sequence with the translation con-

tained in the GPS data. First, a key image is selected in the video sequence at time i. Then, the relative translation and rotation of the camera between the first image and the key image are estimated from the video sequence. The GPS data allow to extract the translation of the camera in the geo-referenced coordinate frame. Finally, an estimation $\tilde{R}_1$ of the rotation matrix R_1 is obtained thanks to the angle between the relative translation and the GPS translation.

2.2. Geometric explanation

Let us consider, in figure 1, the translation $\vec{t_{relative}}$ in the camera coordinate frame and the translation $\vec{t_{GPS}}$ in the Universal Transverse Mercator (UTM) coordinate frame. We can see that the system $\{\mathcal{C}_1; translation; \mathcal{C}_i\}$ is rigid, whatever the reference frame. Therefore, the rotation that aligns the direction of $\vec{t_{relative}}$ over the direction of $\vec{t_{GPS}}$ is also the rotation that aligns the cameras' axes and thus, we can deduce R_1.

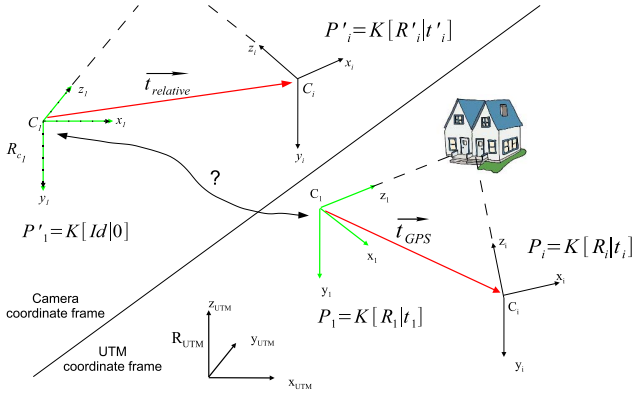


Fig. 1. Matching of the vectors $\vec{t_{relative}}$ and $\vec{t_{GPS}}$ in order to estimate R_1

However, one degree of freedom can not be determined because there is an infinite number of 3D rotations between the vectors $\vec{t_{relative}}$ and $\vec{t_{GPS}}$. This unknown is eliminated by assuming that the rotation's axis is vertical. This assumption is valid in practice since most of the outdoor video sequences have a horizontal aiming axis (within 15 degrees according to [7])

2.3. Computation of $\vec{t_{relative}}$

Let $P'_i = K[R'_i|t'_i]$ be the projection matrix of the camera C_i, relatively to the camera C_1. The change of reference frame between the two cameras is given by:

$$\mathcal{R}_{C_1}\mathcal{T}_{\mathcal{R}_{C_i}} = \begin{bmatrix} R_i'^{-1} & -R_i'^{-1}t'_i \\ 0 & 1 \end{bmatrix} \quad (1)$$

Since $\vec{t_{relative}}$ is the position of C_i expressed in $\mathcal{R}_{C_1}$:

$$\vec{t_{relative}} = C_{i_{\mathcal{R}_{C_1}}} = {}^{\mathcal{R}_{C_1}}\mathcal{T}_{\mathcal{R}_{C_i}} C_{i_{\mathcal{R}_{C_i}}} = -R_i'^{-1}t'_i \quad (2)$$

R'_i and t'_i are estimated with a classical structure from motion method [8].

2.4. Matching of $\vec{t_{relative}}$ and $\vec{t_{GPS}}$

C_i can also be expressed in $\mathcal{R}_{C_1}$ thanks to the GPS data:

$$C_{i_{\mathcal{R}_{C_1}}} = {}^{\mathcal{R}_{C_1}}\mathcal{T}_{\mathcal{R}_{UTM}} C_{i_{\mathcal{R}_{UTM}}} = R_1 C_{i_{\mathcal{R}_{UTM}}} + t_1 \quad (3)$$

because

$$\mathcal{R}_{C_1}\mathcal{T}_{\mathcal{R}_{UTM}} = \begin{bmatrix} R_1 & t_1 \\ 0 & 1 \end{bmatrix} \quad (4)$$

where t_1 can be deduced from:

$$\begin{array}{rc} & C_{1_{\mathcal{R}_{UTM}}} = {}^{\mathcal{R}_{UTM}}\mathcal{T}_{\mathcal{R}_{C_1}} C_{1_{\mathcal{R}_{C_1}}} = -R_1^{-1}t_1 \\ \Leftrightarrow & t_1 = -R_1 C_{1_{\mathcal{R}_{UTM}}} \end{array} \quad (5)$$

Replacing t_1 in equation (3) by equation (5):

$$\begin{array}{rcl} C_{i_{\mathcal{R}_{C_1}}} & = & R_1 C_{i_{\mathcal{R}_{UTM}}} - R_1 C_{1_{\mathcal{R}_{UTM}}} \\ & = & R_1(C_{i_{\mathcal{R}_{UTM}}} - C_{1_{\mathcal{R}_{UTM}}}) = R_1 \vec{t_{GPS}} \end{array} \quad (6)$$

Finally, we obtain from (2) and (6):

$$\vec{t_{relative}} = R_1 \vec{t_{GPS}} \quad (7)$$

The vector $\vec{t_{GPS}}$ is measured between the points (X_i, Y_i, Z_i) and (X_1, Y_1, Z_1) that are extracted from the GPS data, and $\vec{t_{relative}}$ is computed with R'_i et t'_i as explained in the previous section.

2.5. Computation of R_1

We want to estimate R_1 from equation 7. As explained in section 2.2, we eliminate the extra degree of freedom by constraining R_1 to be a rotation in the horizontal plane, giving $\tilde{R}_1$. Therefore, the angle θ between the horizontal components of $\vec{t_{relative}}$ and $\vec{t_{GPS}}$ defines the angle of rotation $\tilde{R}_1$.

The vertical axis is $\vec{y}$ in the camera coordinate frame, and $\vec{z}$ in the UTM coordinate frame. Thus $\tilde{R}_1$ is estimated from $\vec{t_{relative}} = (t_{r_x}, t_{r_y}, t_{r_z})$ and $t_{GPS_x} = (t_{GPS_x}, t_{GPS_y}, t_{GPS_z})$ with:

$$\tilde{R}_1 = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ 0 & 0 & -1 \\ -\sin\theta & \cos\theta & 0 \end{bmatrix} \quad (8)$$

where $\theta = acos(t_{r_x} t_{GPS_x} + t_{r_z} t_{GPS_y})$

3. PRECISE MODEL/IMAGE REGISTRATION

Now that a coarse estimation $(\tilde{R}_1, \tilde{t}_1)$ of the pose is available, it is possible to simultaneously determine a set of 3D/2D feature correspondences and the precise pose of the camera.

3.1. Feature detection and labeling

We decided to use line features because they are more stable than points. Particularly, they are more robust to noise and occlusions. In the image, the lines are detected with a classical method combining the Canny edge detection and the Hough transform. Concerning the 3D model, the building's corners are projected in the image plane, and linked 2-by-2 in order to obtain the building's outline in the image plane.

The number of 3D/2D correspondences must be small enough to avoid a high computational cost and to have a ratio of *outliers* compatible with the RANSAC algorithm. Instead of fixing ad-hoc thresholds, we propose to reduce the number of possible correspondences by labeling the lines. Indeed, the image is classified into 3 regions (sky, building, ground) extracted with the method of geometric context extraction [7]. Then, the lines that are close to the sky/building border or the ground/building border are selected and labeled either *sky* or *ground*. All the vertical lines are selected and labeled *vertical*. Similarly, the lines of the 3D model are also labeled.

Finally, we build the initial 3D/2D correspondences by taking into account only the pairs of lines with the same label. A threshold is used only for the vertical lines to avoid correspondences between vertical lines that are too distant from each other. This threshold is fixed and is equal to $0.5 \times Image\,Width$.

3.2. Robust estimation with RANSAC

RANSAC algorithm enables to iteratively combine correspondences search and pose computation. The method can be summarized as follows:

1. Randomly select a subset of the initial correspondence set.
2. Compute camera pose using this random subset.
3. Test the resulting pose by searching all possible *inliers* in the initial correspondence set. If this pose provides the biggest set of *inliers* then this set is saved.
4. Repeat the steps 1 to 3 N times.
5. Re-compute the precise pose with the biggest set of *inliers*.

The number of iterations N is adaptatively determined with the method described in [8] (p. 117).

The lines are represented in polar coordinates (ρ, θ). In step 3, a correspondence is considered to be an *inlier* if:

$$|\rho_{M3d} - \rho_{Img}| < s_{\rho_{min}} \;\; and \;\; |\theta_{M3d} - \theta_{Img}| < s_{\theta_{min}}$$

The thresholds ($s_{\rho_{min}}$, $s_{\theta_{min}}$) are set to the value of the standard deviation of the measurement error.

3.3. Pose computation

The camera pose is computed during the steps 2 and 4 of the RANSAC algorithm, using a set of correspondences. For this computation, we use the method of virtual visual servoing presented in [9]. This method is based on a control law that minimizes the error between the current lines (projected from the 3D model) and the corresponding desired lines (detected in the image).

If the set of correspondences is well chosen, then there is only one final position of the camera that allows this minimization to be achieved. In theory, 3 lines in a non-degenerated configuration are sufficient, but due to the noise in the images, we need at least 4 lines to achieve a precise computation of the pose.

4. RESULTS

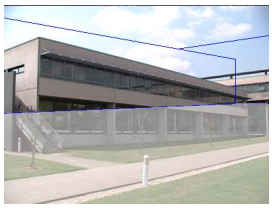
Two video sequences were acquired in the university of Rennes with a hand held camera. The GPS data was collected simultaneously with the video stream. During an experimentation with bad conditions for a GPS acquisition (i.e. next to a building and under cloudy sky), we determined that the horizontal precision of the GPS is about 5m. This horizontal data was linearly interpolated in order to provide one position at each temporal instant corresponding to an image of the video sequence. Moreover, since the vertical precision of the GPS is unreliable, a better estimation was obtained by arbitrarily initializing the camera's vertical position to 1.5m above the ground, where the ground was modeled by a Delaunay triangulation between all the footprints of the buildings.

The parameters of our method are unchanged for all the sequences. The thresholds used to select the *inliers* are $(s^l_\theta, s^l_\rho) = (0.03, 2)$. The choice for the key image is manual for the moment.

Figure 2 shows the results with the sequence *ifsic*. Figure 2.a shows the alignment after the first part of the algorithm. We can see the first image on which the GIS model is superimposed in blue. The semi-transparent surface represents the ground in the GIS. Note that the visible facades of the GIS correspond to those visible in the image. Therefore, the objective of the first part is reached because a set of feature correspondences can now be built.

Figure 2.b shows the final registration obtained after 958 iterations of RANSAC algorithm with an initial set of 19 correspondences. The lines of the image, detected after Canny detection (2.c) and Hough transform and labeling from context extraction (2.d), are shown in Figure 2.e. The lines projected from the 3D model are shown in figure 2.f.

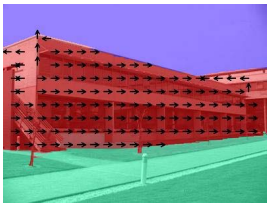
The results validate our approach. The first part provides a good approximation of the pose in spite of the GPS errors, motion estimation errors, and the assumption concerning the camera's aiming axis. In the second part of the method, the main limitation comes from the non-detection of some build-



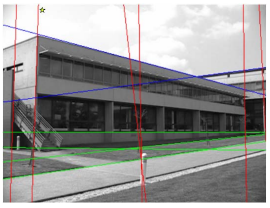
(a) Coarse registration

(b) Final registration

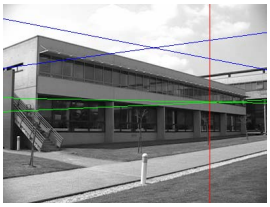
(c) Canny edge detection



(d) geometric context



(e) Lines in the image



(f) Line from the 3D model

Fig. 2. Results for the video sequence *ifsic*

(a) Coarse registration

(b) Final registration

Fig. 3. Results for the video sequence *beaulieu*

ing's edges in the image. Such edges show poor contrast and are close to other lines (e.g the lower edge of the building which is surrounded by windows and shadows). When all building's edges are detected, then the pose is correctly estimated, even with a coarse pose obtain in the first part.

5. CONCLUSION AND FUTURE WORK

This paper presents a method for automatic initial registration of GIS and video data. The solution contains two parts. First, a coarse estimation of the camera pose using the video sequence and the GPS data. Second, a precise estimation of the camera pose using feature line correspondences and RANSAC algorithm. The first results have shown that the coarse estimation of the pose is good enough to enable a search for feature correspondences. Concerning the precise registration, a good alignment was obtained provided that building edges were detected in the image. In the following, we will include the automatic choice of the key image and suppress the assumption concerning the camera aiming axis. We will also improve the feature matching process by using structure from motion techniques.

6. REFERENCES

[1] P.E. Debevec, C.J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach," *Computer Graphics*, vol. 30, no. Annual Conference Series, pp. 11–20, 1996.

[2] S. Teller, M. Antone, Z. Bodnar, M. Bosse, S. Coorg, M. Jethwa, and N. Master, "Calibrated, registered images of an extended urban area," *Int. J. Comput. Vision*, vol. 53, no. 1, pp. 93–107, 2003.

[3] G. Reitmayr and T. Drummond, "Going out: robust model-based tracking for outdoor augmented reality.," in *ISMAR*, 2006, pp. 109–118.

[4] L. Liu and I. Stamos, "Automatic 3d to 2d registration for the photorealistic rendering of urban scenes," in *CVPR '05 - Volume 2*, Washington, DC, USA, 2005, pp. 137–143, IEEE Computer Society.

[5] M.A. Fischler and R.C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[6] P. David, D. DeMenthon, R. Duraiswami, and H. Samet, "Softposit: Simultaneous pose and correspondence determination," in *ECCV (3)*, 2002, pp. 698–714.

[7] D. Hoiem, A. Efros, and M. Hebert, "Geometric context from a single image," in *ICCV*. October 2005, vol. 1, pp. 654 – 661, IEEE.

[8] R.I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[9] E. Marchand and F. Chaumette, "Virtual visual servoing: a framework for real-time augmented reality," *Computer Graphics Forum*, vol. 21(3), pp. 289–298, September 2002.

7. ACKNOWLEDGMENTS

This work was funded by orange labs and Similar Network of Excellence.