Registration with a Zoom Lens Camera for Augmented Reality Applications

Gilles Simon, Marie-Odile Berger

Registration with a Zoom Lens Camera for Augmented Reality Applications

Gilles Simon and Marie-Odile Berger
LORIA- INRIA Lorraine
BP 101
54602 Villers les Nancy, France
email:{gsimon@loria.fr,berger@loria.fr}

Abstract

We focus in this paper on the problem of adding computer-generated objects in video sequences that have been shot with a zoom lens camera. While numerous papers have been devoted to registration with fixed focal length, little attention has been brought to zoom lens cameras. In this paper, we propose an efficient two-stage algorithm for handling zoom changing which are are likely to happen in a video sequence. We first attempt to partition the video into camera motions and zoom variations. Then, classical registration methods are used on the image frames labeled camera motion *while keeping the internal parameters constant, whereas the zoom parameters are only updated for the frames labeled* zoom variations. *Results are presented demonstrating registration on various sequences. Augmented video sequences are also shown.*

Video sequences of our results can be seen at URL `http://www.loria.fr/~gsimon/iwar99.html`.

1. Introduction

Augmented reality (AR) is a technique in which the user's view is enhanced or augmented with additional information generated from a computer model. In contrast to virtual reality, where the user is immersed in a completely computer-generated world, AR allows the user to interact with the real world in a natural way. This explains why interest in AR has substantially increased in the past few years and medical, manufacturing or urban planning applications have been developed [2, 5, 12, 14].

In order to make AR systems effective, the computer generated objects and the real scene must be combined seamlessly so that the virtual objects align well with the real ones. It is therefore essential to determine accurately the location and the optical properties of the cameras. The registration task must be achieved with special care because the human visual system is very good at detecting even small mis-registrations.

There has been much research in the field of vision-based registration for augmented reality [1, 9, 11, 14]. However these works assume that the internal parameters of the camera are known (focal length, size of the pixel, center point) and they only address the problem of computing the pose of the camera. This is a strong limitation of these methods because zoom changing is likely to happen in a video sequence. Recent attempts have been made to cope with varying internal parameters for AR applications [8]. However this approach uses targets arbitrarily positioned in the environment. It is therefore of limited use if outdoor scenes are considered.

In this paper we extend our previous works on vision based registration methods [9, 10] to the case of zoom-lens cameras. Zoom-lens camera calibration is still found to be very difficult for several reasons [13, 3]: modeling a zoom-lens camera is difficult due to optical and mechanical misalignments in the lens system of a camera. Moreover, zoom-lens variations can be confused with camera motions: for instance, it is difficult to discriminate a translation along the optical axis from a zoom.

In this paper, we take advantage of our application field to reduce the problem complexity. Indeed, we assume that the viewpoint and the focal length do not change at the same time. This assumption is compatible with the techniques used by professional movie-makers. We develop in this paper an original statistical approach: for each frame of the sequence, we test the hypothesis of a zoom against the hypothesis of a camera motion. If the motion hypothesis is retained, we still have to compute the camera pose with the old internal parameters. Otherwise, the internal parameters are computed assuming that the camera pose does not change.

This paper is organized as follows: first, we discuss in section 2 the pinhole camera model and we show the difficulties to recover both the camera pose and the internal parameters with varying focal lengths. Section 3 then de-

scribes our original method for zoom/motion partitioning of the sequence. This section also describes how registration is performed from this segmentation. Examples which demonstrate the effectiveness of our method are shown in section 4.

2. Registration difficulties with a zoom-lens camera

In this section, we first describe the pinhole model which is widely used for camera modeling. Then we describe our attempts to compute both the zoom and the motion parameters in a single stage. This task is called full calibration in the following. We show that classical registration methods fail to recover both the internal and the external parameters, even though some of the intrinsic parameters are fixed.

2.1. The pinhole camera model

Let (x, y, z) represent the coordinates of any visible point M in a fixed reference system (world coordinate system) and let (x_c, y_c, z_c) represent the coordinates of the same point in the camera centered coordinate system (Fig. 1). The relationship between the two coordinate systems is given by

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + T = [R\,T] \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

where $[R, T]$ is the 3D displacement (rotation and translation) from the world coordinate system to the camera coordinate system.

We assume that the camera performs a perfect perspective transform with center O at a distance f of the image plane. The projection of M on the image plane is $(f\frac{x_c}{z_c}, f\frac{y_c}{z_c})$. If $1/k_u$ (resp $1/k_v$) is the size of the pixel along the x axes (resp. y axes), its pixel coordinates are:

$$m = (k_u f \frac{x_c}{z_c} + u_0, k_v f \frac{y_c}{z_c} + v_0) \qquad (1)$$

where u_0, v_0 are the coordinates of the principal point of the camera (i.e. the intersection of the optical axis and the image plane).

The coordinates of a 3D point $M = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ in a world coordinate system and its pixel coordinates $m = \begin{pmatrix} u \\ v \end{pmatrix}$ are therefore related by

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} k_u f & 0 & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{A} [R\,T] \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Full camera calibration amounts to compute 10 parameters: 6 external parameters (3 for the rotation and 3 for the translation) and 4 internal parameters ($\alpha_u = k_u f$, $\alpha_v = k_v f$, u_0 and v_0). Internal and external parameters are collectively referred to as camera parameters in the following.

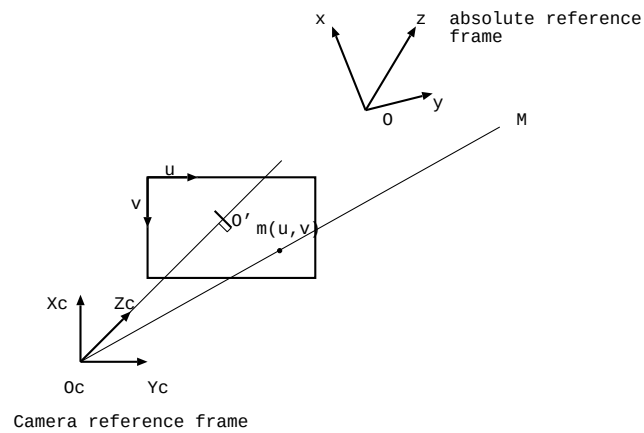


Figure 1. The perspective transformation.

2.2. Direct full calibration

When the internal parameters are computed off-line, the registration process amounts to compute the displacement $[R, T]$ which minimizes the re-projection error, that is the error between the projection of known 3D features in the scene and their corresponding 2D features detected in the image. For sake of clarity, we only suppose that the 3D features are points but we can also consider free form curves [9]. Moreover, we have shown that 2D/2D correspondences can be added to improve the viewpoint computation [10].

The camera pose is therefore the displacement $[R, T]$ which minimizes the reprojection error

$$\min_{R,T} \sum dist(proj(M_i), m_i)^2$$

where minimization is performed only on the external parameters.

Theoretically, zoom-lens variations during shooting can be recovered in the same way. We have therefore to compute not only the camera viewpoint but also the internal

camera parameters (focal length, pixel size, optical center) which minimize the reprojection error.

$$\min_{R,T,\alpha_u,\alpha_v,u_0,v_0} \sum dist(proj(M_i), m_i)^2$$

As mentioned by several authors [3], this approach is unable to recover both the internal and external parameters. To overcome this problem, some authors have proposed to reduce the number of unknowns by fixing some of the internal parameters to predefined values. As several experimental studies proved that the ratio $\frac{\alpha_u}{\alpha_v}$ remains almost constant during zoom variations [4], the set of the internal parameters to be estimated is then reduced to α_u, u_0, v_0. Unfortunately this approach fails to recover the right camera parameters. Consider for instance Fig. 2 which exhibits the results when registration is achieved on the 6 external parameters and the 3 internal parameters. As the house stands on a calibration target, the internal and external parameters can be computed for each frame using classical calibration techniques [6]. They can therefore be compared to those computed with the registration method. The camera motions with respect to the turntable and zoom variations during the *cottage sequence* are shown in Table 1. The camera trajectory along with the focal length computed for each frame of the sequence are shown in Fig. 2 in dotted lines. They have to be compared to the actual parameters which are shown in bold solid lines on the same figure. Note that the trajectory is the position of the camera in the horizontal plane and the arrows indicates the optical axis. These results prove that some camera motions are confused with zoom variations: besides the common confusion between zoom and translation along the optical axis, other motions do not correspond to the actual one: between the frames 13 and 14, a translation is detected and is compensated by a camera zoom out. This can be explained as follows: let $M_i = (x_c^i, y_c^i, z_c^i)$ be the model points expressed in the camera coordinates system and let $m_i = (u_i, v_i)$ be their projections in the image plane. From equation 1, we get the new projections (u_i', v_i') of points M_i after a focal variation Δf or a translation along the optical axis $\Delta \mathbf{T} = (0, 0, \Delta t_z)$ have occured. In the both cases, we obtain:

$$\begin{pmatrix} u_i' \\ v_i' \end{pmatrix} = \begin{pmatrix} u_i \\ v_i \end{pmatrix} + k \begin{pmatrix} u_i - u_0 \\ v_i - v_0 \end{pmatrix},$$

where $k = \Delta f / f$ for the focal change and $k = k_t = -\Delta t_z / z_c^i$ for the translation. k_t depends on the depth of the model points, but if $z_c^i = z_0 + \Delta z_c^i$ where $\Delta z_c^i \ll z_0$ for each model point (that is the object is relatively far from the camera), then it is clear that the translation can be interpreted as a focal change.

Finally, we consider the particular case of the sequences where camera pose and zoom do not change at the same time. This particular case is very interesting for practical applications: indeed, when professional movie-makers make shootings, they generally avoid to mix camera motions and zoom variations. To take advantage of the structure of these sequences, we compute the reprojection error in the two possible cases *zoom alone* and *camera motion alone*: (i) we consider that the internal parameters do not change and we search for the camera pose $[R, T]$ that minimizes the reprojection error (ii) we consider that the camera is fixed and we search for the internal parameters. Surprisingly, experiments we conducted show that the smallest of these two residuals does not always match the right camera parameters: Fig. 3 plots the reprojection error between frames 22 to 35 on a camera zoom sequence. For each frame i, the reprojection error between frame i and frame 20 is computed for the zoom and the motion hypothesis. This allows us to see the influence of the zoom magnitude on the criterion. The results prove that this method fails to recover the right camera parameters unless the magnitude of the zoom variation is high.

image	motion/zoom
$0 \rightarrow 20$	rotation $40°$
$20 \rightarrow 35$	zoom in
$35 \rightarrow 40$	translation 10cm
$40 \rightarrow 55$	zoom out
$55 \rightarrow 65$	rotation $-20°$

Table 1. The camera parameters for the cottage sequence.

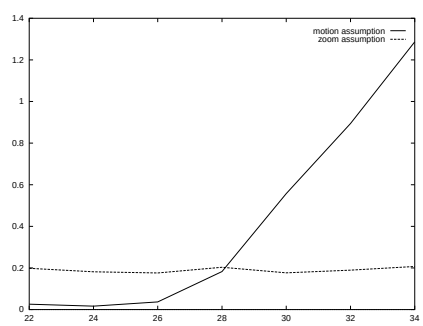


Figure 3. Reprojection error with the zoom and the motion assumption for a camera zoom motion

3. Discriminating between zoom variation and camera motion

The above results show that the classical registration methods cannot be used to cope with zoom-lens cameras. We therefore resort to a two-stage method: we first attempt

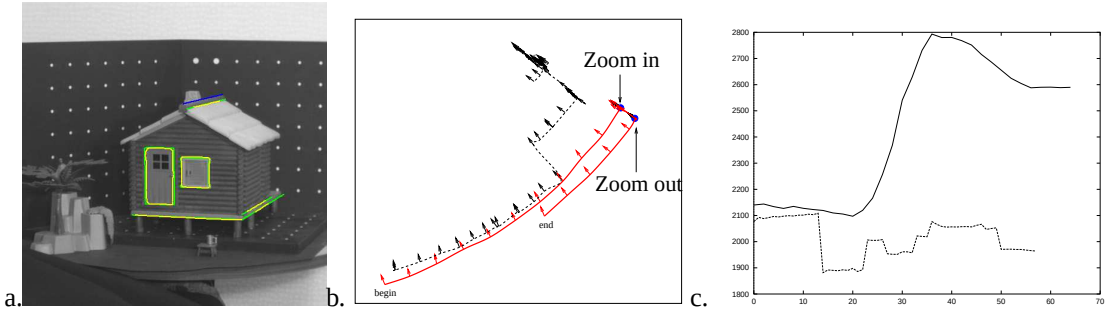


Figure 2. (a) A snapshot of the cottage sequence and the reprojection of the 3D features (b) The actual camera trajectory (bold line) and the computed one (dotted line) (c) the actual (bold line) and the estimated (dotted line) focal length during the sequence

.

to partition the video into camera motions and zoom variations. Then, classical registration methods are used on the image frames labeled *camera motion* while keeping the internal parameters constant, whereas the internal parameters are only computed for the frames labeled *zoom variations*. Unlike other methods for video partitioning which are based on the analysis of the optic flow [15], our method for video partitioning is only based on the analysis of a set of 2D corresponding points which are automatically extracted and matched between two consecutive images. The motion information brought by the key-point is very reliable and allows us to discriminate easily between zoom variation and translation along the optical axis.

Section 3.1 describes the way to extract key-points. Then we present the affine model of a zoom introduced in [4]. Finally we give our algorithm for zoom/motion automatic segmentation of the sequence.

3.1. Extracting and matching key-points

Key-points (or interest points) are locations in the image where the signal changes two dimensionally: corners, T-junctions or locations where the texture varies significantly. We use the approach developed by Harris and Stephens [7]: they exploit the autocorrelation function of the image to compute a measure which indicates the presence of an interest point. More precisely, the eigenvalues of the matrix

$$\begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (I_x = \frac{\partial I}{\partial x} \dots)$$

are the principal curvatures of the auto-correlation function. If these values are high, a key-point is declared.

We still have to match these key-points between two consecutive images. To do this, we use correlation techniques as described in [16].

Fig 4.a and 4.b exhibit the key-points which have been automatically extracted in two successive images in the *cottage scene* and Fig. 4.c shows the matched key-points.

3.2. Modeling zoom-lens cameras

Previous studies on zoom-lens modeling proved that the ratio $\frac{\alpha_u}{\alpha_v}$ is very stable over long time periods. On the contrary, the position of the principal point (u_0, v_0) depends on the zooming position of the camera. This point can vary up to 100 pixels while zooming! However, for most camera lens, it can be shown that the principal point varies on a line while zooming. That is the reason why an affine model with 3 parameters C_0, a_0, b_0 can be used to describe zoom variations. Enciso and Vieville [4] show that if (u', v') and (u, v) are corresponding points after zooming, we have

$$\begin{aligned} u' &= C_0 u + a_0 \\ v' &= C_0 v + b_0 \end{aligned}$$

The current matrix of the internal parameters A' is therefore deduced from the previous one A by:

$$A' = \begin{pmatrix} C_0 & 0 & a_0 \\ 0 & C_0 & b_0 \\ 0 & 0 & 1 \end{pmatrix} A.$$

and the perspective matrix after zooming is deduced from the previous one by the relation:

$$P' = \begin{pmatrix} C_0 & 0 & a_0 \\ 0 & C_0 & b_0 \\ 0 & 0 & 1 \end{pmatrix} P.$$

3.3. Zoom/motion partioning

In this section, we present our approach for zoom/motion partioning. For each frame of the sequence, we test the hy-

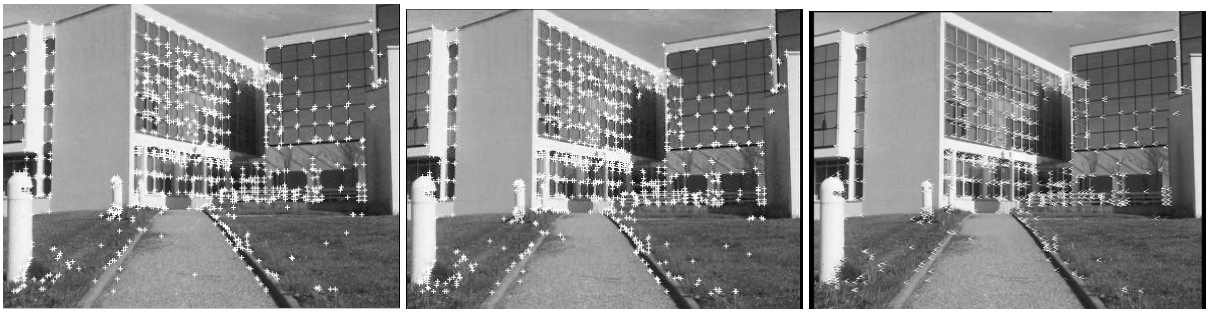


Figure 4. (a,b) :key-points extracted in two consecutive frames (c): the matched key-points.

pothesis of a zoom against the hypothesis of a camera motion. We proceed as follows: key-points $(u_i, v_i)_{\{1 \leq i \leq N\}}$ and $(u'_i, v'_i)_{\{1 \leq i \leq N\}}$ are extracted and matched in two consecutive frames I_k and I_{k+1}. If we suppose that a zoom occurs, the model parameters C_0, a_0, b_0 which best fit the set of corresponding key-points are computed by minimizing the residual

$$r = \frac{1}{N} \sum_{i=1}^{N} (u'_i - C_0 u_i - a_0)^2 + (v'_i - C_0 v_i - b_0)^2. \quad (2)$$

We must now estimate the goodness of fit of the data to the affine model of the zoom. We have to test if the discrepancy r is compatible with the noise magnitude on the extracted key-points. Otherwise the zoom hypothesis should be questioned.

Statistical tests, such as χ^2 tests, are often used to estimate the compatibility of the data with the model with a given significance level a (90% for instance). However, the standard deviation is needed for each datum. In our case, it is very difficult to calculate an error on the location of the key points. The χ^2 test has also a serious drawback: how can we set the significance level a? For a very large value of a, the hypothesis is always admitted, while for a very small value of a the hypothesis is always rejected.

That is the reason why we resort to another criterion to assess the zoom hypothesis. An important thing to note is that a zoom variation does not introduce new features in the images whereas translation motion does: some features which are visible for a camera viewpoint are no longer visible for a neighboring camera position. In Fig. 5.a, point A is not visible from C_k because it is occluded by the object O_1. But point A becomes visible when the camera moves from C_k to C_{k+1}. Note that such a phenomenon also arises for translation along the optical axis (Fig. 5.b). These features which become visible due to the camera motion are very important for assessing the zoom hypothesis. As key-points are not necessarily detected in the areas which become visible or which disappear, the key-points are not well suited for zoom assessment.

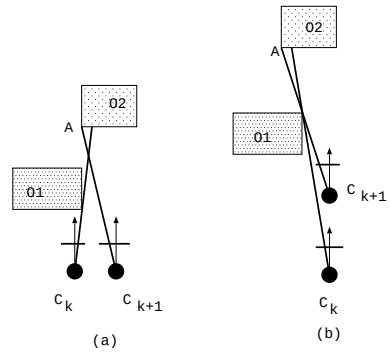


Figure 5. New features appear under translating motion: point A is not visible from C_k but becomes visible from C_{k+1}.

We therefore use the set of all the contours detected in image I_k to assess the parameters (if $C_0 < 1$ we use image I_{k+1}). We first compute a correlation score for each contour. This score belongs to $[-1, 1]$ and is all the better that the zoom hypothesis is fulfilled. If the zoom hypothesis is satisfied, the gray levels $I_k(u, v)$ and $I_{k+1}(C_0 u + a_0, C_0 v + b_0)$ must be nearly the same. Moreover the neighborhood of these two corresponding points must be similar. We therefore use the correlation score to evaluate the zoom hypothesis. First, we define the correlation for a given point $m = (u, v)$ in I_k:

$$score(m) = \frac{\sum_{i,j=-n}^{i,j=n} I_k(u+i, v+j) \times I_{k+1}(C_0(u+i) + a_0, C_0(v+j) + b_0)}{(2n+1)^2 \sigma(I_k) \sigma(I_{k+1})},$$

where $\sigma(I_k)$ (resp. $\sigma(I_{k+1})$) is the standard deviation of I_k (resp. I_{k+1}) at point (u, v) in the neighborhood $(2n+1) \times (2n+1)$ of (u, v) (resp. $(C_0 u + a_0, C_0 v + b_0)$). The score ranges from -1 for two correlation windows which are not similar at all, to 1 for two correlation windows which are identical.

If a contour is given by the points $m_1, ..., m_p$, the score of a contour $\mathcal{C}$ is defined as the average of the scores of all

points:

$$score(\mathcal{C}) = 1/p \sum_{i=1}^{i=p} score(m_i).$$

Finally the score of the *zoom hypothesis* is computed as the minimum of the score of each contour. This is a robust way to assess the zoom hypothesis. Indeed, if a zoom variation really happens, the score is high for each contour, and the global score is high too. On the contrary, if a camera motion happens, the score is generally low for nearly all the contours when the camera moves because the affine zoom model does not match the image transformation. Moreover, in case of a translating motion, the score is low for the contours of I_k which are occluded in I_{k+1}. Hence the global score is low too.

We still have to choose a threshold Th_{score} which allows us to distinguish between zoom variation and camera motion according to the global score. This value has been determined experimentally on various sequences. Experiments we have conducted (see section 4.2) prove that the value $Th_{score} = .5$ can be used for all the considered sequences to discriminate between zoom variation and camera motion even for the difficult case of a translation along the optical axis. Hence, if $global\_score > .5$, the zoom hypothesis is accepted, otherwise the camera motion hypothesis is retained.

3.4. Registration with a zoom lens camera

Once the zoom/motion partitioning has been achieved, registration can be performed from 2D/3D correspondences. As described in [9], we use curve correspondences. Once the curves corresponding to the 3D features have been detected in the first frame of the sequence, they are tracked from frame to frame. If the frame belongs to a camera zoom sequence, then registration is performed only on the set of the internal parameters. Otherwise, registration is performed only on the set of the external parameters. Hence, the camera parameters in frame $k+1$ are deduced from the camera parameters in frame k by the relation:

`if a zoom variation is detected :`

$R^{k+1} = R^k, T^{k+1} = T^k,$

$C_0^{k+1}, u_0^{k+1}, v_0^{k+1} = \underset{C_0,u_0,v_0}{argmin} \sum_i dist(proj(M_i), m_i)^2$

$\alpha_u^{k+1} = C_0^{k+1}\alpha_u^k$

$\alpha_v^{k+1} = C_0^{k+1}\alpha_v^k$

`if a camera motion is detected :`

$\alpha_u^{k+1} = \alpha_u^k, \alpha_v^{k+1} = \alpha_v^k$

$u_0^{k+1} = u_0^k, v_0^{k+1} = v_0^k$

$R^{k+1}, T^{k+1} = \underset{R,T}{argmin} \sum_i dist(proj(M_i), m_i)^2$

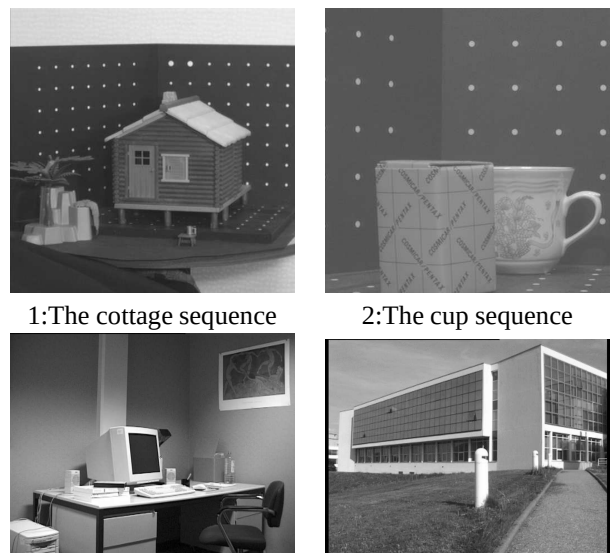
4. Experimental results

In this section, we first justify experimentally the use of the threshold $Th_{score} = 0.5$ to discriminate between zoom variations and camera motions. Then, section 4.2 present results of the partitioning process. Finally, registration results are given and augmented scenes are shown.

4.1. Choosing Th_{score}

To prove that $Th_{score} = 0.5$ is well suited to discriminate between camera motion and zoom variation, we considered a variety of video sequences (see Fig. 6). Each sequence alternates zoom variations with camera motions, including translations along the optical axis T_Z, which are difficult to distinguish from zoom variations. For each frame of the sequence, the labeling in terms of *zoom variation, rotation motion, translation motion* is known. This allows us to compare the results of our algorithm with the actual ones.

[!h]



1:The cottage sequence 2:The cup sequence

3:The office sequence 4:The Loria sequence

Figure 6. Snapshots of the scenes used for testing the zoom/motion partitioning algorithm.

We first compute the score of the zoom hypothesis for each frame of the four sequences. Then we compute the mean along with the standard deviation of the score for the frames of the sequence corresponding to *zoom variation*, T_Z *translation*, *rotation*. These results are shown in table 2: the first column shows the kind of variation undergone by the camera. The second and third columns give the scene under consideration and the number of frames in the sequence corresponding to the camera variation. Columns 4 and 5 show the mean and the standard deviation of the

variation in the camera parameters	scene	nb frames	r	σ_r	mean score	σ_{score}
Zoom	1	6	0.617	0.030	0.747	0.055
	2	4	0.460	0.266	0.860	0.055
	3	32	0.860	0.057	0.677	0.133
	4	29	0.515	0.014	0.561	0.064
Translation along the optical axis	1	2	0.651	0.020	0.393	0.066
	2	4	0.841	0.018	0.274	0.035
	3	16	1.380	0.190	0.047	0.277
Rotation + translation	1	10	3.593	1.439	-0.591	0.171
Panoramic motion	4	15	0.630	0.066	-0.209	0.315

Table 2. Score of the zoom hypothesis for various camera parameters.

residual computed from the corresponding key-points (see equation 2). Finally, columns 6 and 7 shows the mean and the standard deviation of the score of the *zoom hypothesis*. These results clearly show that the use of the residual defined in equation (2) does not permit to discriminate between zoom variations and translation along the optical axis. On the contrary, the score we have defined gives high values when zoom happens and much smaller results when camera motion happens, even in case of T_Z translation. Finally, these experiments prove that the value $Th_{score} = .5$ is appropriate to distinguish zoom variations from camera motions.

4.2. Results in zoom/motion partitioning

We now give detailed results of our algorithm on the *cottage sequence* and the *Loria sequence*. Note that the camera parameters are known for the *cottage sequence* because the house stands on a calibration target. The *Loria sequence* is a long sequence which has been shot outside our laboratory. This sequence consists of 700 frames of size 768×576. The actual camera parameters are not available for this sequence. However we have manually partition the sequence (see table 3) to enable comparison with the automatic algorithm.

For each of the two sequences (Fig. 7), we show the scores computed along the sequence, the results of our partitioning algorithm, and the computed zoom factor C_0. Also shown in the Fig. 7.b and 7.e is the actual partition of the sequence for comparison. For the *cottage sequence*, the algorithm performance is quite good and the computed parameters are very close to the actual parameters. For the *Loria sequence*, the reader can notice that some camera parameters are mis-labeled during panoramic motions between frames 0 and 100 and between frames 600 and 700 (Fig. 7.d). This failure can be easily explained: in the panoramic section, the camera rotation is small and the observed scene is rather far from the camera. Then, the motion induced in the image is close to a translation and the computed C_0 is very close to 1. The affine zoom model is therefore theoretically fulfilled. Fortunately, zoom motion and translation parallel to the image plane can easily be distinguished. Indeed, for a zoom motion, the invariant point of the affine model $(\frac{a}{1-C_0}, \frac{b}{1-C_0})$ is the principal point of the camera and lies approximately in the middle of the image. On the contrary, for a translating motion, this point is outside the image and goes to infinity. Hence the zoom hypothesis is retained if the score is greater than .5 and if the invariant point $(\frac{a}{1-C_0}, \frac{b}{1-C_0})$ lies inside the image. In Fig. 7.a and 7.d, the condition on the invariant point is shown with dotted lines: the value 1 indicates that the invariant point is inside the image, while the value 0 indicates that the invariant point is outside the image. Using these two conditions, the results of the partition process is very good (Fig. 7.b and 7.e).

4.3. Registration results

In this section, registration results are shown for the *cottage sequence* and the *Loria sequence*. As the actual parameters are known for the *cottage sequence*, Fig. 8 and Fig. 9 show the trajectory and the focal length computed with our algorithm (dotted lines) along with the actual parameters (bold lines). The reader can notice that the parameters obtained are in close agreement with the actual values. To prove the accuracy of the camera parameters, we have augmented the scene with a palm tree and a beach umbrella (Fig. 10). Note that the shadows between the scene and the computer generated objects greatly improve the realism of the composite images. They have been computed from a rough 3D reconstruction of the scene given by the corresponding key-points. The reprojection of the 3D model features with the computed camera parameters is also shown. The overall impression is very good.

We do not have the actual camera parameters for the *Loria sequence*. Hence looking at the reprojection of the

Image frames	camera parameters
$0 \rightarrow 120$	panoramic motion
$121 \rightarrow 344$	Zoom in
$345 \rightarrow 408$	no motion, nor zoom
$409 \rightarrow 600$	Zoom out
$601 \rightarrow end$	panoramic motion

Table 3. Camera parameters during the Loria sequence

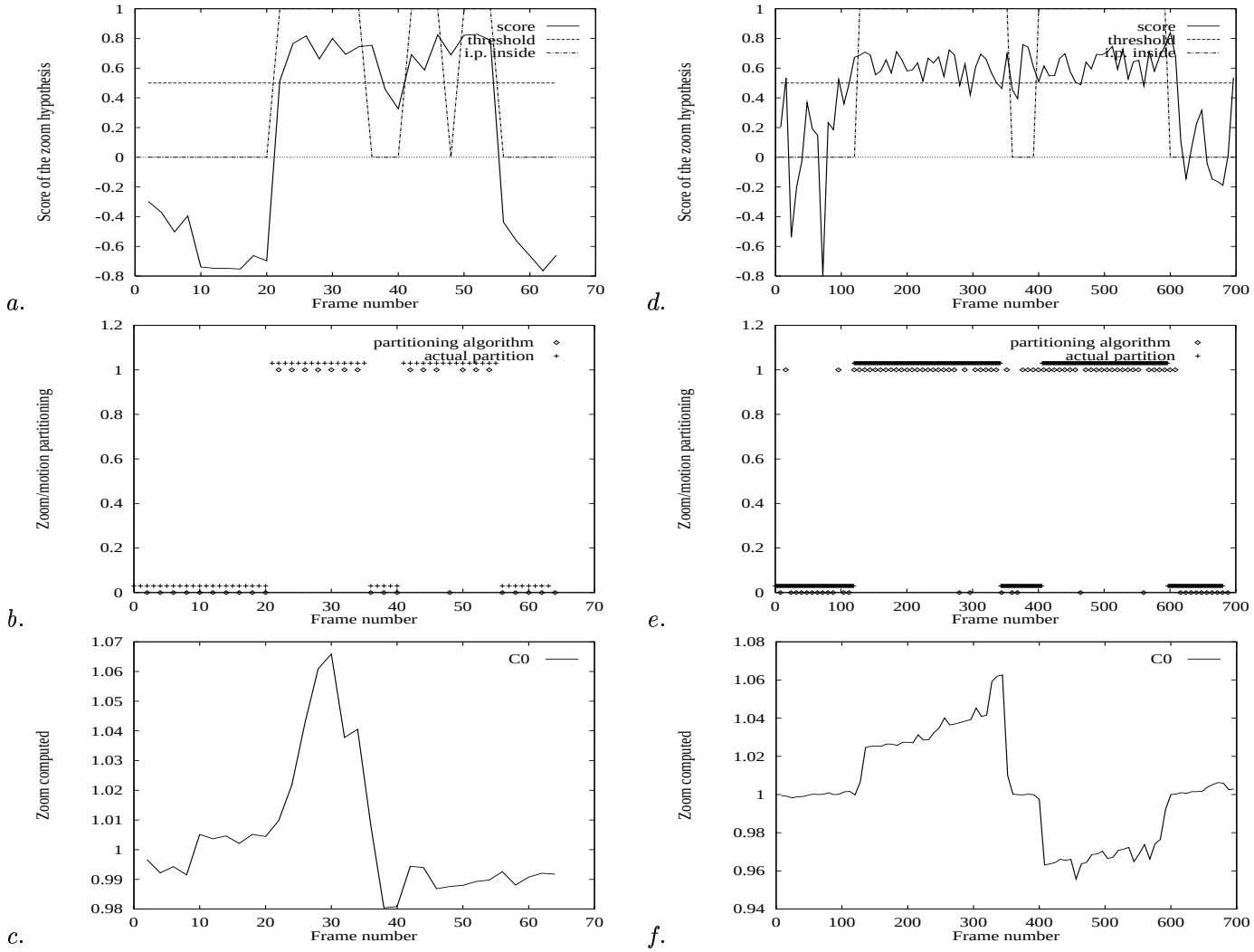


Figure 7. Results for the cottage sequence (left column) and the Loria sequence (right column)

model features is a good way to assess the registration accuracy. Fig. 11 exhibits the reprojection of the model every hundred frames. The reader can notice that the reprojection error is small even at the end of the sequence, which proves the efficiency of our algorithm. Finally, we augment the sequence with the well known sculpture *La femme à la chevelure défaite* realized by *Mirõ*. Note that video sequences of our results can be seen at URL `http://www.loria.fr/~gsimon/iwar99.html`.

5. Conclusion

In this paper we have presented an efficient registration algorithm for a zoom lens camera. We restricted our study to the case of image sequences which alternate zoom variation alone and camera motion alone. This is a quite reasonable assumption which is always fulfilled by professional movie-makers. The performance of our algorithm is quite good and our algorithm is capable of discriminating between zoom variations and T_Z translations. However, our experiments show that some improvements and extensions can be made to our approach.

First, experiments on the *Loria sequence* show that the camera trajectory is somewhat jagged . Smoothing the trajectory afterwards is not appropriate because the correspondences between the image and the 3D model are not maintained. We currently investigate methods to incorporate regularity constraints on the trajectory inside the registration process.

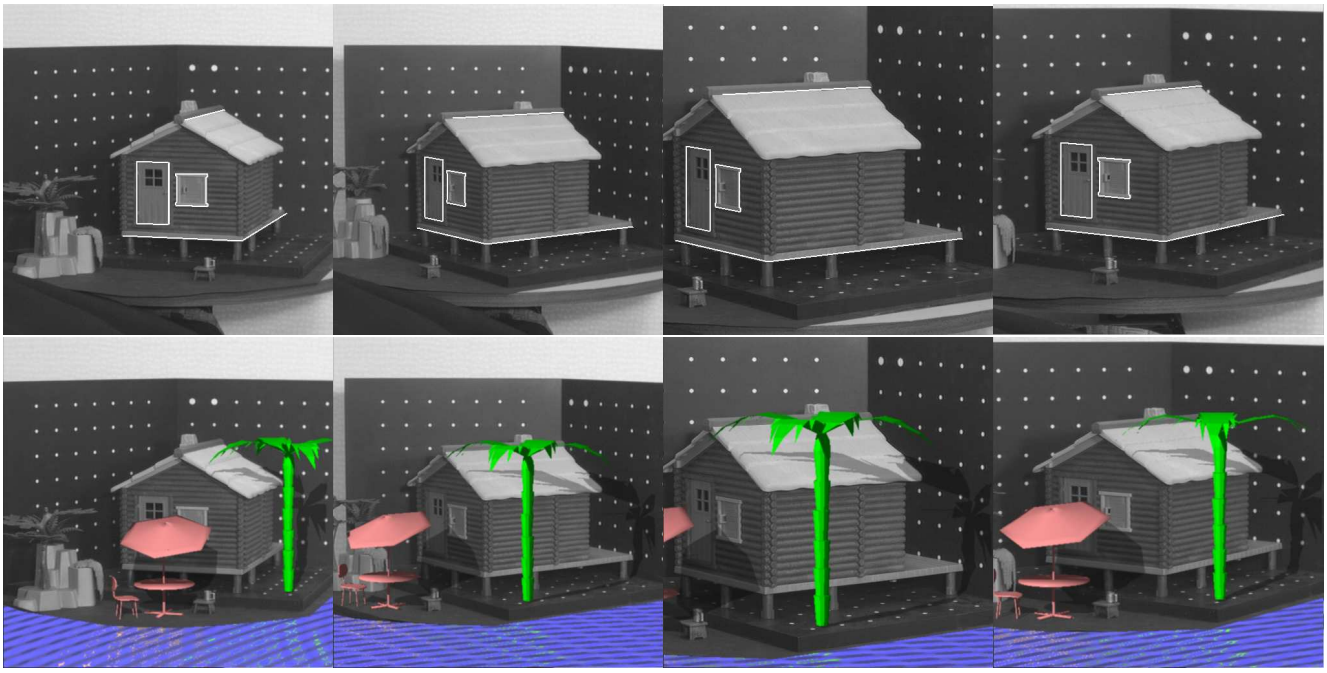


Figure 10. Registration results on the cottage sequence: reprojection of the model (first row) Snapshots of the augmented scene (second row).

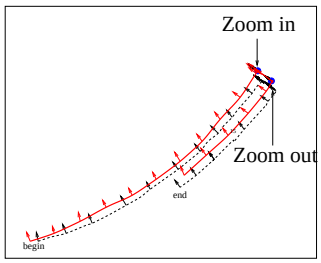


Figure 8. Comparison of the actual trajectory with the computed one.

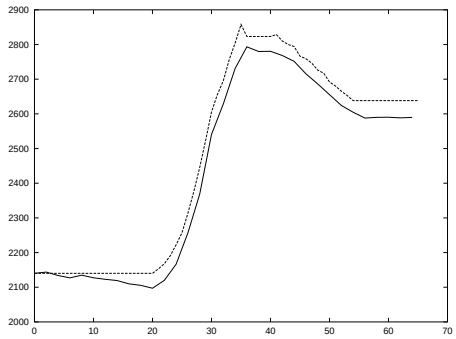


Figure 9. Comparison of the actual focal length α_u (bold lines) with the computed one (dotted lines).

Second, as was observed in our experiments, moving objects in the scene may perturb the partitioning process. Indeed, the correlation score is always low for moving objects and this may lead to false rejection of the zoom hypothesis. Detecting moving objects in the scene prior to the registration process could help to solve this problem.

References

[1] R. T. Azuma and G. Bishop. Improving static and dynamic registration in an optical see through display. *Computer Graphics*, pages 194–204, July 1994.

[2] M.-O. Berger, C. Chevrier, and G. Simon. Compositing Computer and Video Image Sequences: Robust Algorithms for the Reconstruction of the Camera Parameters. In *Computer Graphics Forum, Conference Issue Eurographics'96, Poitiers, France*, volume 15, pages 23–32, Aug. 1996.

[3] S. Bougnoux. From Projective to Euclidiean Space under any Practical Situation, a Criticism of Self-calibration. In *Proceedings of 6th International Conference on Computer Vision, Bombay (India)*, pages 790–796, Jan. 1998.

[4] R. Enciso and T. Vieville. Self-calibration from four views

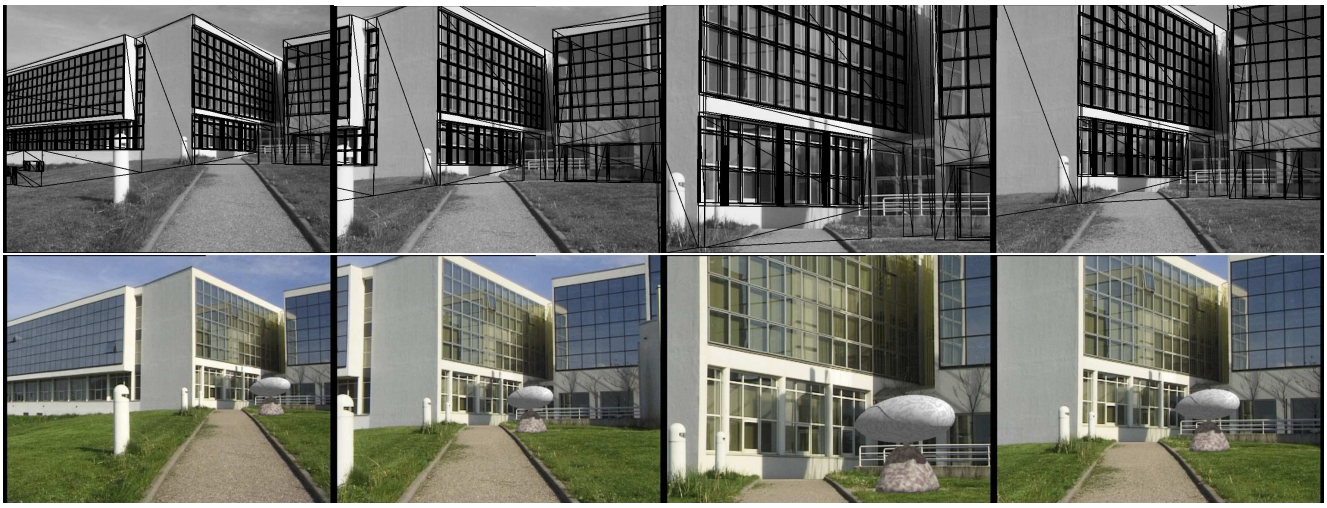


Figure 11. Registration results on the Loria sequence: the reprojection of the model every hundred frames (first row) snapshots of the augmented scene (second row) .

with possibly varying intrinsic parameters. *Image and Vision Computing*, 15(4):293–305, 1997.

[5] G. Ertl, H. Müller-Seelich, and B. Tabatabai. MOVE-X: A System for Combining Video Films and Computer Animation. In *Eurographics*, pages 305–313, 1991.

[6] O. D. Faugeras and G. Toscani. The Calibration Problem for Stereo. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL (USA)*, pages 15–20, 1986.

[7] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *Proceedings of 4th Alvey Conference*, Cambridge, Aug. 1988.

[8] J. Mendelsohn, K. Daniilidis, and R. Bajcsy. Constrained Self-Calibration for Augmented Reality Registration. In *First International Workshop on Augmented Reality, San francisco, USA*, 1998.

[9] G. Simon and M.-O. Berger. A Two-stage Robust Statistical Method for Temporal Registration from Features of Various Type. In *Proceedings of 6th International Conference on Computer Vision, Bombay (India)*, pages 261–266, Jan. 1998.

[10] G. Simon, V. Lepetit, and M.-O. Berger. Computer Vision Methods for Registration: Mixing 3D Knowledge and 2D Correspondences for Accurate Image Composition. In *First International Workshop on Augmented Reality, San francisco, USA*, 1998.

[11] A. State, G. Hirota, D. Chen, W. garett, and M. Livingston. Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking. In *Computer Graphics (Proceedings Siggraph New Orleans)*, pages 429–438, 1996.

[12] A. State, M. Livingstone, W. Garett, G. Hirota, M. Whitton, and E. Pisan. Technologies for Augmented Reality Systems: Realizing Ultrasound Guided Needle Biopsies. In *Computer Graphics (Proceedings Siggraph New Orleans)*, pages 439–446, 1996.

[13] P. Sturm. Self Calibration of a moving Zoom Lens Camera by Pre-Calibration. In *British Machine Vision Conference, Edinburgh, Scotland*, pages 675–684, 1996.

[14] M. Uenohara and T. Kanade. Vision based object registration for real time image overlay. *Journal of Computers in Biology and Medecine*, 1996.

[15] W. Xiong and J. Lee. Efficient scene change detection and camera motion annotation for video classification. *Computer Vision and Image Understanding*, 71(2):166–181, 1998.

[16] Z. Zhang, R. Deriche, O. Faugeras, and Q. Luong. A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry. *Artificial Intelligence*, 78:87–119, Oct. 1995.