Motion Panoramas

Adrien Bartoli, Navneet Dalal, and Radu Horaud
INRIA Rhône-Alpes
655, avenue de l'Europe
38330 Montbonnot Saint-Martin, FRANCE
Corresponding author: Radu Horaud
Radu.Horaud@inrialpes.fr
telephone/fax: +33 476 615 226 / +33 476 615 454
http://www.inrialpes.fr/movi/people/Horaud/

17 February 2003

Abstract

In this paper we describe a method for analysing video sequences and for representing them as mosaics or panoramas. Previous work on video mosaicking essentially concentrated on static scenes. We generalise these approaches to the case of a rotating camera observing both static and moving objects where the static portions of the scene are not necessarily dominant, as it has been often hypothesised in the past. We start by describing a robust technique for accurately aligning a large number of video frames under unknown camera rotations and camera settings. The alignment technique combines a feature-based method (initialisation and refinement) with rough motion segmentation followed by a colour-based direct method (final adjustment). This precise frame-to-frame alignment allows the dynamic building of a background representation as well as an efficient segmentation of each image such that moving regions of arbitrary shape and size are aligned with the static background. Thus a *motion panorama* visualises both dynamic and static scene elements in a geometrically consistent way. Extensive experiments applied to archived videos of track-and-field events validate the approach.

Keywords: video mosaicking, panoramic visualisation, layered representation, motion segmentation, background substraction, texture alignment

1 Introduction

Motion panoramas are visual representations of motion. Traditionally motion is visualised using an image sequence, or a video. Consider, for example the case of a person moving over several tens of meters. A video of such a moving person is gathered, in general, with a rotating and zooming camera such that: (i) the observed person remains in the camera's field of view and (2) preferably at constant image resolution. A compact and convenient representation of such a video is to stitch the individual images into a unique wide-angle panoramic image – a panorama, which is also called a mosaic.

The case of rigid scenes has been thoroughly investigated and a number of methods, algorithms, and software packages are available to produce *static panoramas*. The case that we want to study in this paper is more complex. Indeed, the combination of non-rigid scenes (scenes with multiple and/or articulated moving objects) with camera motion, as well as changes in camera parameters (focus and zoom) raises new difficulties.

The first and main difficulty is to segment each image into regions corresponding to distinct observed motions: multiple object motions and camera motion. The second difficulty is to estimate the camera internal parameters as well as the camera motion parameters such that the mapping from each individual image in the sequence to a single panoramic image can be performed. The third difficulty is to produce a high-quality motion panorama which is basically composed of two layers: a dynamic layer which corresponds to the moving objects and a static layer which corresponds to the static background.

The concept of motion panorama is best illustrated on Figure 1. From an original video (top) two layers are extracted. The static layer, or the background, is used to estimate the camera motion and the camera parameters associated with every image in the sequence. Next a background panorama is built (middle). Finally each individual image is compared to the background panorama in order to extract image regions corresponding to motion – the dynamic layer. Finally, the static and dynamic layers are combined together to form a motion panorama (bottom).

Panoramic photography has received growing interest since a decade [2, 3, 11, 12, 20, 22, 25, 15, 16] resulting in a number of commercial products such as [18]. The idea behind these methods is that there exist a simple invertible transformation between images gathered with a camera rotating around its center of projection [9]. A vast majority of papers (see [20] for a review), concentrates on the static case. While high-quality results are obtained, this assumption prunes many real-life image sequences.

Others have addressed the problem of analysing sequences of one or several moving objects with a static camera [13, 19, 24]. A current approach to detect motion with a static camera is to segment the image into two categories or two layers: a static layer and a dynamic layer, where a layer is a set of pixels. Practical approaches to layered segmentation is background substraction based on pixel-to-pixel comparison between a pre-stored background image and the current image. Of course, these methods work well when background is available.

Methods for analysing videos of moving objects with a moving camera are presented in [14], [12], and [6]. Both are interesting attempts to dynamically build a background image and to find moving object by substracting the background from each individual frame. In [12] the authors propose the use of a direct method to find the camera motion parameters, align frames based on these parameters, and spotting the image regions which do not satisfy these parameters. In [6] block-matching motion detection is first applied to find a motion vector field and this field is clustered to segment dominant motion regions. A direct method (see below) is applied to these regions in order to estimate camera motion. We found that these approaches work well when the moving objects correspond to relatively small image regions. When large portions of the images are occupied by moving objects, direct methods fail to find the camera motion. It is worthwhile to point out that a young company, Dartfish, commercializes software for producing motion panoramas from videos [5]. Their panorama building procedure requires manual intervention both for building the background and for selecting dynamic objects to be eventually overlayed onto the background.

The most crucial characteristics of methods associated with motion panorama construction are (i) the ability to deal with large dynamic image regions and (ii) the accuracy in frame alignment. Generally speaking, two categories of methods are available: Feature-based methods [22] and direct methods [11]. The former consists in extracting image features such as points of interest, matching such features over several images, and estimating the mapping between images based on feature-to-feature correspondences. The latter consists of finding the image-to-image mapping which best aligns the image intensity values (or red, green, and blue values for colour images).

Feature-based techniques belong to an interesting class of methods which allow the estimation of an image mapping with as few as two feature-to-feature assignments (see below) and which may be combined with outlier rejection techniques. Therefore these methods can successfully be used to estimate the image mapping corresponding to camera motion while throwing out portions of the image which correspond to a different motion. Nevertheless, the process of extracting features from images introduces artifacts such as offsets in feature localisation. Moreover, features are observed only in the presence of important changes in image intensities. Detected features are not homogeneously distributed across the images which may cause alignment problems.

Direct methods consist of finding the mapping between images by minimising the discrepancy between their pixel values and/or colours, i.e., image correlation techniques. These methods produce the best results in terms of image alignment and hence in terms of the final quality of the mosaic, provided that a good initialisation is available. Robust correlation techniques were suggested in the past, i.e., correlation in the presence of artifacts. However, the idea of combining correlation under such image deformations as plane-projective transformations with robust techniques is not a realistic one.

Both the feature-based and the direct methods outlined above contribute to estimate the image-to-image transformations necessary for aligning the input images onto the panoramic image output. Another important ingredient is the segmentation of each image into two layers, a static one associated with camera motion and a dynamic one associated with moving scene objects. If camera motion has been estimated, one may detect the presence of moving objects by warping two images in the sequence and by detecting intensity or colour discrepancies at each pixel location. Such a method provides a fair initialisation of image regions corresponding to moving objects but fails to provide reliable results for producing motion panoramas. Indeed, consider complex human motion such as running: At any short-time interval some body parts move while some other body parts remain still and homogeneous regions such as skin or cloth are detected only along their contours. For these reasons, a more sophisticated motion detection technique is required.

This paper has the following original contributions:

- First we describe a three step method for building a static panorama in the presence of multiple moving objects. With respect to previous methods, we allow for large objects. (i) We suggest a parameterisation for zooming cameras with two rotational degrees of freedom, pan and tilt. Under the assumption that the focal length smoothly varies over a long image sequence but is almost constant between two consecutive frames we show that the planar homography allowing for frame-to-frame alignment needs only two points to be matched. Therefore the performance of any feature-based method using a robust estimator is improved both in terms of reliability and efficiency. (ii) Based on this initial camera motion estimation we warp the previous and next frames onto the current frame in order to detect moving regions. This three-frame motion detector optimistically detects these regions thus minimising the risk that outliers are included in the static layer. (iii) We describe an efficient direct method which aligns pixels in between two frames based on colour constancy and by minimising over four parameters, three rotational degrees of freedom and focal length. Unlike previous methods, we carefully design the error function such that the most time-consuming processes (such as computing the image gradients) are carried out outside the inner loop of the iterative minimisation procedure;
- Second we describe a background/foreground segmentation method. The static panorama accounts for a background image that is being built dynamically as the video proceeds. Each pixel in this image has statistics associated with it thus allowing simple and reliable comparison with each individual image. Since no assumption is made about the number of objects, the number of motions, etc., highly deformable and/or articulated objects such as humans can be easily detected, and
- Third we describe extensive tests made with 350-frame videos of track-and-field events (high jump and pole vault). We demonstrate that camera parameters may be reliably estimated from noisy, low-resolution archived VHS footage. High quality motion panoramas are produced in spite of the poor quality of the input data.

1.1 Paper organisation

The remainder of the paper is organised as follows. Section 2 introduces the un-calibrated camera motion parameterisation. Section 3 formulates the problems of finding image alignments with features and by direct comparison of pixel colours. Section 4 describes the feature-based method and section 5 describes the direct image alignment method. Section 6 summarises the motion panorama algorithm and describes in detail the dynamic background substraction method allowing the final segmentation of each image into background and foreground. Experiments and their results are shown in section 7 and conclusions and directions for future work are discussed in section 8. Appendix A analyses the sensitivity of the alignment to camera calibration errors and appendix B derives a simple formula useful for the incremental estimation of a homography.

2 Mathematical preliminaries and notations.

We consider the pin-hole camera model which projects the 3-dimensional world onto a 2-dimensional image, represented at each time instant i by a 3×4, rank 3, homogeneous matrix $\mathbf{P}_i$. We express all 3-D entities in a standard camera coordinate frame with its orientation at the first time instant. It is assumed that the camera rotates around its optical center and therefore there is no translation associated with camera motion. The 3×3 matrix $\mathbf{K}_i$ contains the intrinsic parameters of the camera at time i and the 3×3 matrix $\mathbf{R}_i$ defines its orientation. One can write $\mathbf{P}_i \simeq \begin{bmatrix} \mathbf{K}_i\mathbf{R}_i & \mathbf{0} \end{bmatrix}$, where '$\simeq$' denotes equality up to a scale factor.

Let m_{ij} be an image point (the j-th point in the i-th image) and let the 2-vector $\boldsymbol{m}_{ij}$ designate its pixel coordinates while the 3-vector $\boldsymbol{q}_{ij}$ designates its homogeneous image coordinates. This is the 2-D projection of the 3-D point M_j whose homogeneous world coordinates are denoted by the 4-vector $\boldsymbol{Q}_j$: $\boldsymbol{q}_{ij} = \mathbf{P}_i\boldsymbol{Q}_j$. We denote by $\Psi()$ the non-linear function mapping homogeneous image coordinates onto pixel coordinates, $\boldsymbol{m} = \Psi(\boldsymbol{q})$. That is, if the 3-vector $\boldsymbol{q}$ has coordinates q_1, q_2, and q_3, $\Psi(\boldsymbol{q}) = (q_1/q_3,\ q_2/q_3)^\top$. The coordinates of the ray passing through this point are given by $\boldsymbol{r}_{ij} \simeq (\mathbf{K}_i\mathbf{R}_i)^{-1}\boldsymbol{q}_{ij}$. This may well be interpreted as a point at infinity R_j with homogeneous world coordinates $\boldsymbol{R}_{ij}^\top = (\boldsymbol{r}_{ij}^\top\ 0)$. It is now possible to derive the inter-frame projective model, e.g., between frame i and frame k for a point j, see figure 2:

$$
\begin{aligned}
\boldsymbol{q}_{kj} &\simeq \mathbf{K}_k\mathbf{R}_k\boldsymbol{r}_{ij} \\
&\simeq \mathbf{K}_k\mathbf{R}_k\mathbf{R}_i^{-1}\mathbf{K}_i^{-1}\boldsymbol{q}_{ij} \\
&\simeq \mathbf{K}_k\mathbf{R}_{ki}\mathbf{K}_i^{-1}\boldsymbol{q}_{ij}
\end{aligned}
$$

Matrix:

$$\mathbf{H}_{ki} \simeq \mathbf{K}_k\mathbf{R}_{ki}\mathbf{K}_i^{-1} \tag{1}$$

defines a homography and the equation above fixes its parameterisation under the assumption that the camera undergoes a rotational motion around its fixed center of projection. One may use the Rodrigues equation to parameterise the rotation $\mathbf{R}_{ki}$ undergone by the the camera: $\mathbf{R}_{ki} = \mathbf{I} + \sin\phi_{ki}[\boldsymbol{\omega}_{ki}]_\times + (1 - \cos\phi_{ki})[\boldsymbol{\omega}_{ki}]_\times^2$, where $[\boldsymbol{\omega}]_\times = (d\mathbf{R}/dt)\,\mathbf{R}^\top$, (the tangent operator) is a skew-symmetric matrix.

We consider the pinhole camera model. Traditionally there are 4 parameters associated with such a model: the focal length, the aspect ratio, and the pixel coordinates of the center of projection. The aspect ratio is fixed by the video standard being used and therefore is known. Throughout the paper we will assume that the center of projection lies at the image center. Appendix A analyses the error associated with this approximation. Only the focal length is unknown and it varies with time: Let f_i be the focal length at time i. Matrix $\mathbf{K}_i$ can now be written as a diagonal matrix:

$$\mathbf{K}_i = \begin{bmatrix} f_i & 0 & 0 \\ 0 & f_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

An useful assumption is that the rotation angle ϕ is small in between two consecutive frames, i and $i+1$. With the approximations $\sin\phi \approx \phi$ and $\cos\phi \approx 1$ we obtain $\mathbf{R} = \mathbf{I} + \phi[\boldsymbol{\omega}]_\times = \mathbf{I} + [\boldsymbol{\Phi}]_\times$ and finally a *small* homography writes, i.e., eq. (1):

$$\mathbf{H}_{i+1,i} = \begin{bmatrix} f_{i+1} & 0 & 0 \\ 0 & f_{i+1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -\phi^z_{i+1,i} & \phi^y_{i+1,i} \\ \phi^z_{i+1,i} & 1 & -\phi^x_{i+1,i} \\ -\phi^y_{i+1,i} & \phi^x_{i+1,i} & 1 \end{bmatrix} \begin{bmatrix} 1/f_i & 0 & 0 \\ 0 & 1/f_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

We consider a sequence of m frames. The parameters associated with each frame are $\mathcal{M}_i \equiv (\mathbf{R}_i, \mathbf{K}_i)$. The layer segmentation consists of a binary classification of pixels lying in the dynamic layer $\mathcal{F}_i$. The parameters of each frame are defined by $\boldsymbol{\theta}_i \equiv (\mathcal{M}_i, \mathcal{F}_i)$. The background image is denoted by $\mathcal{B}$. The complete parameter set is $\boldsymbol{\theta} \equiv \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m, \mathcal{B}\}$, also denoted by $\boldsymbol{\theta} \equiv \{\boldsymbol{\theta}_i, \mathcal{B}\}$. Images are denoted by $\mathcal{I} \equiv \{\mathcal{I}_1, \ldots, \mathcal{I}_m\}$. Throughout the paper there are three colours associated with each pixel, therefore $\mathcal{I}_i(\boldsymbol{m})$ is a 3-valued function of two variables, the image coordinates of $\boldsymbol{m}$.

3 Problem formulation

We cast the motion panorama construction problem as the problem of finding the registration parameters $\hat{\boldsymbol{\theta}}$ that best explain the images $\mathcal{I}$, or equivalently, we look for:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \Pr(\mathcal{I}|\boldsymbol{\theta}),$$

where $\Pr(\mathcal{I}|\boldsymbol{\theta})$ is the probability of the images, given the registration. This is equivalent to maximising the likelihood of the registration under the assumption of uniform probability on the registration and the scene.

Unfortunately, solving this problem is, in general, intractable. The solution is often approximated by assuming the conditional independence of non-consecutive frames, which leads to a frame-to-frame registration, i.e. $m-1$ lower-order problems, this is the *direct* method:

$$\Pr(\mathcal{I}|\boldsymbol{\theta}) \approx \prod_{i=1}^{i=m-1} \Pr(\mathcal{I}_i, \mathcal{I}_{i+1}|\boldsymbol{\theta}_i, \boldsymbol{\theta}_{i+1}), \tag{4}$$

i.e., $\hat{\boldsymbol{\theta}} \approx \{\arg\max_{\boldsymbol{\theta}_i} \Pr(\mathcal{I}_i|\boldsymbol{\theta}_i)\}$.

Another possibility to solve for $\hat{\boldsymbol{\theta}}$ is to reduce the amount of information contained in the images by selecting sets of salient features $\mathcal{Q}$ and computing the registration which best explains these features, the *feature-based* method:

$$\hat{\boldsymbol{\theta}} \approx \arg\max_{\boldsymbol{\theta}} \Pr(\mathcal{Q}|\boldsymbol{\theta}). \tag{5}$$

This problem has a practical solution, often referred to as *bundle adjustment*, which most of the time makes the assumption that the noise on feature positions is independent, identically distributed, and Gaussian. It lies in the class of *feature-based* methods. It can be solved using non-linear optimisation techniques. The previous assumption of conditional independence for non-consecutive frames may be used to compute an initial guess for the registration.

4 Feature-based image alignment

As mentioned above, the practical solution to solve for eq. (5) is bundle adjustment which is a non-linear minimisation technique applied to an error function based on the geometric distance between image points and image point predictions [23]. The latter necessitates proper initialisation. In this section we describe a robust method to find initial estimates for the matrices $\mathbf{H}_{i+1,i}$ and a bundle adjustment method to refine these estimates.

The robust method uses the MSAC algorithm [21] which is based on the RANSAC algorithm [7]. This requires point matches between consecutive images and splits the matches into two categories: inliers and outliers. The bundle adjustment method considers all the homographies with all their associated matches (inliers) and attempts to improve their estimates by minimising a sum of squares of geometric distances.

Points of interest (features) are extracted from each image using the Harris corner detector. These points are tracked over the image sequence in order to establish point-to-point correspondences. This point tracker uses standard correlation techniques without any prior motion model. It finds matches associated with all motions (camera and scene objects) but is unable to assess the reliability of these matches.

4.1 Two-point image alignment

First we consider the problem of estimating a homography between two consecutive frames, $\mathbf{H}_{i+1,i}$, from as few point matches as possible. A linear solution may be obtained by minimising the following algebraic distance [8]:

$$\sum_{j=1}^{n} d_{\text{alg}} \left(\boldsymbol{q}_{i+1,j}, \mathbf{H}_{i+1,i} \boldsymbol{q}_{i,j} \right)^2 = \|\mathbf{A}\boldsymbol{h}\|^2$$

where $d_{\text{alg}}(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} \times \boldsymbol{y}\|$ and n is the number of point matches $\boldsymbol{q}_{i+1,j} \Leftrightarrow \boldsymbol{q}_{i,j}$ available to solve the problem. Vector $\boldsymbol{h}$ is formed with the 8 entries of the homography matrix and $\mathbf{A}$ is a $2n \times 8$ measurement matrix. In the general case one needs a minimum of $n = 4$ point matches to solve the problem. Since homography estimation lies in the inner loop of robust estimators based on random sampling and since the number of trials depends on n, it is desirable to maintain the latter as low as possible.

Consider a camera mounted on a tripod. In this case one may assume that there is no rotation around the optical axis: $\phi^z = 0$. Moreover, one may assume that the focal length does not vary too much between two consecutive images: $f_{i+1} = f_i$ (note that this does not mean at all that the focal length is constrained to be constant through the whole video sequence). Under these assumptions, the image-to-image homography is approximated by:

$$\mathbf{H}_{i+1,i} = \begin{bmatrix} 1 & 0 & f_i \phi^y_{i+1,i} \\ 0 & 1 & -f_i \phi^x_{i+1,i} \\ -\phi^y_{i+1,i}/f_i & \phi^x_{i+1,i}/f_i & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & h_1 \\ 0 & 1 & h_2 \\ h_3 & h_4 & 1 \end{bmatrix} \tag{6}$$

Each point match contributes with two rows in the measurement matrix $\mathbf{A}$. Since there are only four unknowns, $\boldsymbol{h} = (h_1\ h_2\ h_3\ h_4)^\top$, one needs 2 point matches in order to estimate a homography. The solution in this case is reduced to inverting the 4×4 matrix $\mathbf{A}$. Estimates for $\phi^x_{i+1,i}$, $\phi^y_{i+1,i}$, and f_i from the estimated entries of $\mathbf{H}_{i+1,i}$ are easily obtained as well, [8].

4.2 Robust estimation

A robust estimator is necessary because of the presence of both spurious matches and matches corresponding to moving objects and not to camera motion. We want to deal with such practical situations where 50% or more of the observed points belong to the static background. This high rate of outliers immediately rules out M-estimators and LMedS methods. The method of choice is RANSAC which was proposed in 1981 [7] and extensively described in [8] for homography estimation.

A variant of RANSAC is MSAC [21]. RANSAC searches for the motion parameters for which the number of inliers is maximised. In [21] it is argued that this may lead to poor estimation of the parameters and it is proposed to maximise a cost function which sums up the individual errors induced by the camera motion on each feature correspondence. MSAC will provide us with a set of inliers: point matches satisfying the camera motion model and hence corresponding to the background. An important feature of this method is that it is able to throw out a large number of outliers, the number of outliers could be much larger than the number of inliers.

MSAC consists in sampling a minimal set of correspondences (2 in our case), estimating the corresponding motion, and computing its score. The score is computed based on the relative error induced by the camera motion on each point correspondence (see below).

This process is iterated a number of times in order to guarantee a probability of success given a lower bound on the fraction of data contaminated by outliers. The highest-score motion parameters are thus selected. An efficient implementation is obtained by dynamically reducing the number of iterations needed each time the estimated motion is improved, i.e., each time the lower bound on outliers is reduced.

4.3 Maximum likelihood estimation

Up to now we estimated frame-to-frame correspondences. Global correspondences (across many consecutive frames) are required in order to gather all image points corresponding to the same 3-D ray and optimally estimate the homographies. The bundle adjustment method described below considers all matched points and minimises a geometric error in order to optimally determine the 3-D rays under the constraint that they pass through the fixed center of projection.

In the previous paragraphs an algebraic error was minimised which lead to a simple, linear solution that could efficiently be casted into an outlier rejection procedure. It is therefore possible to initialise a maximum likelihood estimator, which minimises a physically meaningful error. This is what bundle adjustment does, [23]. It consists in minimising the re-projection error, defined as the discrepancy between the rays and their matching image points. The minimisation is conducted over all the homographies (parameterised by angles and focal lengths) and all the 3-D rays.

More specifically the error to be minimised is:

$$\sum_{j=1}^{n}\sum_{i=1}^{m}\delta_{ij}d_{\text{geom}}\left(\Psi(\mathbf{K}_i\mathbf{R}_i\boldsymbol{r}_{ij}),\Psi(\boldsymbol{q}_{ij})\right)^2$$

where n is the total number of inliers, m is the number of images being considered, and δ_{ij} is an entry of an association matrix $\boldsymbol{\Delta}$ which is equal to 1 if inlier j is present in image i and equal to 0 otherwise.

In the case of independent and identical Gaussian noise on feature positions, bundle adjustment is known to yield the maximum likelihood estimate. In [20] it is proposed to use such a technique to minimise the difference between 3D rays and not between re-projected features. The former may induce a bias in the estimate when compared to the latter which is optimal with respect to the physical meaning of the error. We use the Levenberg-Marquardt algorithm to conduct the optimisation and compute the Jacobian matrix via finite differences [17], which we found to be as fast as the analytic differentiation which is generally preferred. An efficient implementation is obtained by considering the special sparse structure of the normal equations to be solved.

5 Direct image alignment

As already mentioned, direct image alignment consists of finding the homography $\mathbf{H}$ minimising:

$$E=\sum_{\boldsymbol{q}\in\mathcal{I}_2}\|\mathcal{I}_2(\Psi(\boldsymbol{q}))-\mathcal{I}_1(\Psi(\mathbf{H}\boldsymbol{q}))\|^2 \tag{7}$$

where $\mathcal{I}_1$ and $\mathcal{I}_2$ are the coloured images corresponding to two frames in the video sequence, $\boldsymbol{q}$ is the homogeneous 3-vector associated with a pixel in the first frame and in the second frame. Finding the homography $\mathbf{H}$ that minimises eq. (7) solves for the direct method class of solutions, i.e., eq. (4).

Given a current estimate of the homography, $\mathbf{H}_{\text{old}}$, we attempt to find an incremental improvement, $\mathbf{H}_{\text{imp}}$ such that the newly estimated homography becomes (this is illustrated in Figure 3):

$$\mathbf{H}_{\text{imp}}\mathbf{H}_{\text{new}}=\mathbf{H}_{\text{old}} \tag{8}$$

Under this representation, the error to be minimised becomes:

$$E=\sum_{\boldsymbol{q}\in\mathcal{I}_2}\|\mathcal{I}_2(\Psi(\mathbf{H}_{\text{imp}}\boldsymbol{q}))-\mathcal{I}_1(\Psi(\mathbf{H}_{\text{old}}\boldsymbol{q}))\|^2 \tag{9}$$

In Appendix B the following formula for $\mathbf{H}_{\text{imp}}$ is derived:

$$\mathbf{H}_{\text{imp}}=\begin{bmatrix}1-f & -\phi^z & f_2\phi^y\\ \phi^z & 1-f & -f_2\phi^x\\ -\phi^y/f_2 & \phi^x/f_2 & 1\end{bmatrix} \tag{10}$$

In this equation f denotes the increment to be estimated allowing to update the focal length from its "old" value, f_2, to its new value, $f_2(1+f)$, and ϕ^x, ϕ^y, and ϕ^z denote the angular increments allowing to update the rotational part of the homography. Therefore, there are four parameters to be estimated:

$$\boldsymbol{\theta} = \begin{pmatrix} \phi^x & \phi^y & \phi^z & f \end{pmatrix}^\top$$

Notice that the entries of $\mathbf{H}_{\text{imp}}$ are linear in $\boldsymbol{\theta}$ and one may write:

$$\mathbf{H}_{\text{imp}} = \mathbf{H}(\boldsymbol{\theta})$$

In order to solve the minimisation problem stated above, we describe $\mathcal{I}_2$ by its first order Taylor expansion around the vector parameter $\boldsymbol{\theta}$ at $\boldsymbol{\theta} = 0$

$$\mathcal{I}_2(\Psi(\mathbf{H}(\boldsymbol{\theta})\boldsymbol{q})) = \mathcal{I}_2(\Psi(\mathbf{H}(\mathbf{0})\boldsymbol{q})) + \left(\frac{d\mathcal{I}_2}{d\Psi}(\Psi(\mathbf{H}(\mathbf{0})\boldsymbol{q}))\right)^\top \left[\frac{d\Psi}{d\boldsymbol{\theta}}(\mathbf{H}(\boldsymbol{\theta} = \mathbf{0})\boldsymbol{q})\right] \boldsymbol{\theta}$$

Noticing that $\mathbf{H}(\mathbf{0}) = \mathbf{I}$ and with $\boldsymbol{m} = \Psi(\boldsymbol{q})$ we obtain:

$$\mathcal{I}_2(\Psi(\mathbf{H}(\boldsymbol{\theta})\boldsymbol{q})) = \mathcal{I}_2(\boldsymbol{m}) + \left(\frac{d\mathcal{I}_2}{d\boldsymbol{m}}\right)^\top \left[\frac{d\Psi}{d\boldsymbol{\theta}}(\mathbf{H}(\boldsymbol{\theta} = \mathbf{0})\boldsymbol{q})\right] \boldsymbol{\theta}$$

Eq. (9) can now be approximated by:

$$E_{\text{approx}} = \sum_{\boldsymbol{m} \in \mathcal{I}_2} \left\| \left(\frac{d\mathcal{I}_2}{d\boldsymbol{m}}\right)^\top \left[\frac{d\Psi}{d\boldsymbol{\theta}}(\mathbf{H}(\boldsymbol{\theta} = \mathbf{0})\boldsymbol{q})\right] \boldsymbol{\theta} - \boldsymbol{K}(\boldsymbol{m}) \right\|^2 \tag{11}$$

with:

$$\boldsymbol{K}(\boldsymbol{m}) = \mathcal{I}_1(\Psi(\mathbf{H}_{\text{old}}\boldsymbol{q})) - \mathcal{I}_2(\boldsymbol{m})$$

The 2-vector $\boldsymbol{g}(\boldsymbol{m}) = \left(\frac{d\mathcal{I}_2}{d\boldsymbol{m}}\right)$ is the image gradient at pixel location $\boldsymbol{m}$ (for colour images there are 3 such gradients for the red, green, and blue components) and $\left[\frac{d\Psi}{d\boldsymbol{\theta}}(\mathbf{H}(\boldsymbol{\theta} = \mathbf{0})\boldsymbol{q})\right]$ is the Jacobian associated with the mapping of $\boldsymbol{m}$ with homogeneous coordinates $\boldsymbol{q} = (q_1\ q_2\ 1)^\top$:

$$\mathbf{J}(\boldsymbol{m}) = \left[\frac{d\Psi}{d\boldsymbol{\theta}}(\mathbf{H}(\boldsymbol{\theta} = \mathbf{0})\boldsymbol{q})\right] = \begin{bmatrix} -\frac{q_1 q_2}{f_2} & f_2 + \frac{q_1^2}{f_2} & -q_2 & -q_1 \\ -f_2 - \frac{q_2^2}{f_2} & \frac{q_1 q_2}{f_2} & q_1 & -q_2 \end{bmatrix}$$

With the notation $\boldsymbol{b}^\top(\boldsymbol{m}) = \boldsymbol{g}^\top(\boldsymbol{m})\mathbf{J}(\boldsymbol{m})$ the error above becomes:

$$\begin{aligned} E(\boldsymbol{\theta}) &= \sum_{\boldsymbol{m} \in \mathcal{I}_2} \left(\boldsymbol{b}^\top(\boldsymbol{m})\boldsymbol{\theta} - \boldsymbol{K}(\boldsymbol{m})\right)^\top \left(\boldsymbol{b}^\top(\boldsymbol{m})\boldsymbol{\theta} - \boldsymbol{K}(\boldsymbol{m})\right) \\ &= \boldsymbol{\theta}^\top \left(\sum \boldsymbol{b}(\boldsymbol{m})\boldsymbol{b}^\top(\boldsymbol{m})\right) \boldsymbol{\theta} - 2\left(\sum \boldsymbol{K}^\top(\boldsymbol{m})\boldsymbol{b}^\top(\boldsymbol{m})\right) \boldsymbol{\theta} + \sum \boldsymbol{K}^\top(\boldsymbol{m})\boldsymbol{K}(\boldsymbol{m}) \end{aligned}$$

The Euler condition $\frac{dE(\boldsymbol{\theta})}{d\boldsymbol{\theta}} = 0$ yields the following constraint for reaching a minimum:

$$\left(\sum (\mathbf{J}^\top(\boldsymbol{m})\boldsymbol{g}(\boldsymbol{m})\boldsymbol{g}^\top(\boldsymbol{m})\mathbf{J}(\boldsymbol{m}))\right) \boldsymbol{\theta} = \sum (\boldsymbol{K}^\top(\boldsymbol{m})\boldsymbol{g}^\top(\boldsymbol{m})\mathbf{J}(\boldsymbol{m}))$$

Matrix $\mathbf{A} = \sum (\mathbf{J}^\top(\boldsymbol{m})\boldsymbol{g}(\boldsymbol{m})\boldsymbol{g}^\top(\boldsymbol{m})\mathbf{J}(\boldsymbol{m}))$ is a definite, positive, and symmetric 4×4 matrix.

To summarise, the algorithm allowing the alignment of two images is the following:

1. Estimate the gradient vector $\boldsymbol{g}(\boldsymbol{m})$ at each point $m \in \mathcal{I}_2$;
2. Initialise $\mathbf{H}_{\text{old}}$ using the feature-based method;
3. While(true);

   For each point $m \in \mathcal{I}_2$ compute $K(\boldsymbol{m})$ and $\mathbf{J}(\boldsymbol{m})$;

   (a) Find a solution for the unknown vector $\boldsymbol{\theta}$
   (b) Compute $\mathbf{H}_{\text{imp}}$;
   (c) $\mathbf{H}_{\text{old}} = \mathbf{H}_{\text{imp}}^{-1}\mathbf{H}_{\text{old}}$;
   (d) If $E < \varepsilon$ then terminate, else continue;

It is worthwhile to notice that, with this formulation, the image gradient (step 1) is estimated only once. Unlike previous methods [20] both the focal length and the rotational angles are estimated simultaneously within the same linear solution (step 3-a).

6 Building motion panoramas

The classical method for producing panoramas with a rotating camera is to combine the individual images into a unique image using the previously estimated homographies. In the case of a moving (rotating) camera observing a dynamic scene the panorama building strategy is more complex because one has to combine panorama building with motion segmentation.

Methods for producing panoramic images from dynamic scenes were already suggested in the past. However these methods make the assumption that the background is predominant.

The overall method suggested below summarizes as follows:

1. The feature-based method is applied in order to find initial estimates of camera motion and camera settings, i.e., matrices $\mathbf{H}_{i,i-1}$;
2. Each image is compared, pixel by pixel, with its previous and next images in the sequence, the latter two images being warped using $\mathbf{H}_{i,i-1}$ and $\mathbf{H}_{i+1,i}^{-1}$. Pixels are classified into two classes: foreground and background. Foreground pixels are farther grouped into connected regions;
3. Apply the direct method to background regions associated with the images in the sequence in order to refine the previously estimated matrices $\mathbf{H}_{i,i-1}$ and to obtain a better alignment;
4. Form a background panoramic image – for each pixel in this image gather colour information from all the images contributing to this pixel, and
5. Map the previously computed foreground regions onto the background panorama, compare the region and the background, refine the foreground regions, and form a motion panorama.

Steps 1 and 3 were described in detail above. The remainder of this section describes in detail steps 2 (initial background/foreground segmentation), 4 (building the background panorama), and 5 (building the motion panorama)

6.1 Initial background/foreground segmentation

A rough segmentation of each image into two layers, static foreground and dynamic foreground is trivial for static cameras. For moving cameras the process is more complex because camera motion, most often, compensates for a rapidly moving object such that this object remains in the center of the image. An initial estimate of camera motion being provided, it is now possible to map the previous and the next images onto the current one, and directly compare the colours at each pixel location. This is done by the following distance function:

$$\boldsymbol{d}\left(\mathcal{I}_i(\boldsymbol{m}), \mathcal{I}'_{i-1}(\boldsymbol{m})\right)$$

where $\mathcal{I}'_{i-1}$ is obtained from:

$$\mathcal{I}'_{i-1}(\boldsymbol{m}) = \mathcal{I}_{i-1}(\Psi(\mathbf{H}_{i,i-1}\boldsymbol{q}))$$

Notice that $\boldsymbol{d}$ computes the colour discrepancy between the two pixels as they appear in two consecutive images. In practice the Mahalanobis distance is used and the distance becomes:

$$\boldsymbol{d}^2\left(\mathcal{I}_i(\boldsymbol{m}), \mathcal{I}'_{i-1}(\boldsymbol{m})\right) = \left(\mathcal{I}_i(\boldsymbol{m}) - \mathcal{I}'_{i-1}(\boldsymbol{m})\right)^\top \mathbf{C}^{-1}\left(\mathcal{I}_i(\boldsymbol{m}) - \mathcal{I}'_{i-1}(\boldsymbol{m})\right)$$

The covariance matrix $\mathbf{C}$ is estimated using the red, green, and blue colour values of all pixels and for every image in the video sequence. It was experienced elsewhere [1] that, under general imaging conditions and without a prior colour model, this is one of the most reliable distance measures in colour space. Moreover, the distance function just described can be easily normalised to return values in the interval $[0, 1]$, i.e., d_{N}.

Based on this normalised Mahalanobis distance we define the likelihood of a pixel to belong to the foreground:

$$p_{\boldsymbol{m}} = \max\left(\boldsymbol{d}_{\mathrm{N}}\left(\mathcal{I}_i(\boldsymbol{m}), \mathcal{I}'_{i-1}(\boldsymbol{m})\right), \boldsymbol{d}_{\mathrm{N}}\left(\mathcal{I}_i(\boldsymbol{m}), \mathcal{I}'_{i+1}(\boldsymbol{m})\right)\right) \tag{12}$$

Indeed, a large likelihood of foreground $p_{\boldsymbol{m}}$ means that there is a large colour discrepancy between three consecutive images at a pixel $\boldsymbol{m}$ and that this pixel does not obey the estimated camera motions, $\mathbf{H}_{i,i-1}$ and $\mathbf{H}_{i+1,i}$. Therefore it is possible to compute $p_{\boldsymbol{m}}$ values at each pixel j of an image i and obtain a probability of foreground associated with each location – a probability image. However, a lot of pixel-level noise and false detections are present. False detections are due to large homogeneous regions (the interior of these regions are not detected as moving regions) as well as to complex motions such as articulated body motions. Indeed, such motions are characterised by the fact that some body parts move while some other body parts remain still. In order to remove this type of artifacts and since a foreground object always exists as a region, the probability image is smoothed using a Gaussian kernel.

Since the intention is to optimistically detect foreground regions and increase the probability that the remaining pixels belong to the background, we apply to $p_{\boldsymbol{m}}$ a threshold that is less than half the value of the maximum of the $p_{\boldsymbol{m}}$'s. A pixel is classified as foreground, if there is enough colour discrepancy between either images $\mathcal{I}_i$, $\mathcal{I}_{i-1}$ OR $\mathcal{I}_i$, $\mathcal{I}_{i+1}$.

6.2 Building background and motion panoramas

Once the pixels in each image are classified as just explained, image-to-image transformations are refined using the direct method applied to background pixels. We recall that this technique optimally estimates the homographies associated with camera motion by directly estimating the underlying parameters: the angles of rotation and the focal length.

To build a background panorama we consider each potential pixel in its image plane (in practice the background image is cylindrical). For each one of these pixels we gather the contributions from the video sequence. Let k be the number of consecutive video frames contributing to the same pixel in the background panorama. In our experiments the angular speed of the camera is of approximatively 0.005 radians per frame (or 0.3^0). At this speed there are, on an average, 20 frames contributing at each background pixel. Notice that this is a relatively fast motion, 7.5^0 per second. Therefore it is significant to gather statistics over these 20 frames.

In detail, let $\boldsymbol{b}$ be a pixel of the panorama and let $\mathcal{B}(\boldsymbol{b})$ be the final colour associated with this pixel. We gather k colours from k contributing video frames, $\mathcal{B}_l$, $1 \leq l \leq k$. Let μ and σ be the mean and standard deviation associated with these k colours. Assuming a Gaussian distribution we determine a weight α_l associated with each contribution:

$$\alpha_l = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{(\mathcal{B}_l - \mu)^2}{2\sigma^2}\right)$$

By proper, normalisation, $\sum \alpha_l = 1$ we obtain the colour of a background pixel:

$$\mathcal{B}(\boldsymbol{b}) = \sum_{l=1}^{k} \alpha_l \mathcal{B}_l$$

Once a background panoramic image has thus been created one can warp each individual frame onto this image plane and perform background substraction to extract the foreground. Since the previously estimated foreground regions encompass moving scene objects, one can reliably use these regions for final foreground estimation.

7 Experimental results

The motion mosaicking technique described above was successfully applied to a number of video sequences of track-and-field events. The initial data correspond to archived VHS (analog) recordings which are digitised such that each individual frame in the sequence has 354 by 280 pixels. A typical high-jump or pole-vault sequence contains approximatively 350 frames. The results shown below were produced when the method was applied to one out of four frames, i.e., a sequence of 85 frames. The reason for sub-sampling the number of frames in the sequence is because the camera motion between two consecutive frames is too small and therefore the estimation of the camera motion becomes numerically unstable. Knowledge about the video standard being used (PAL in our case) and about the digitisation process allows one to consider square pixels (the aspect ratio is 1) but we have no knowledge about the remaining camera parameters. We assume that the center of projection lies at the image center (see Appendix A). We also assume that the camera is mounted onto a tripod which does not necessarily imply that the camera rotates around its center of projection; the motion parallax is however small in this case. We have no a priori information about the zoom or about the camera angular motion (pan and tilt).

Figure 4 shows a set of results obtained with one frame in a sequence. It is worthwhile to notice that the number of inliers is eventually lower than the number of outliers. In order to appreciate the improvement in mosaic alignment, Figure 5 shows a panorama of the background using the feature-based method (left) and the same panorama using the direct method (right).

The complete background and motion panoramas for this high-jump video are shown on Figure 6. The pan and tilt camera angular values as well as the zoom settings are shown of Figure 7 as they are estimated by the direct method (frame-to-frame alignment).

The next example shows a pole-vault video sequence. Figure 8 shows a sample of 6 original frames (frames 134, 173, 262, 322, 362, and 402) together with the extracted foreground regions. The background and motion panoramas are shown on Figure 9.

8 Conclusions

In this paper we presented a new method for analysing videos of dynamic scenes and of representing them as motion panoramas. One of the most crucial steps within the process of building such a panorama is the alignment of the images in the sequence in the presence of dominant moving objects. Indeed, real-life video footage such as track-and-field events contain multiple and complex moving objects. We designed a frame-to-frame alignment method that proceeds gradually in three steps. The image alignment technique starts with a feature-based method for initialising the transformation parameters associated with both the camera motion and its zoom setting, and for eliminating image features that do not satisfy these parameters. Based on these estimates a rough segmentation of each image is then performed and, finally, a direct method adjusts the parameters such that the image intensities are finely aligned in between consecutive frames. This fine alignment is key to the success of the dynamic construction of a background panorama representation and of the final segmentation of each individual image into foreground and background. Eventually, foreground regions from several images are combined with the background panorama into a geometric-consistent manner to form a motion panorama.

Motion panoramas encapsulate more information that the information required for visualising them. For example in [4] it is shown how to recover 3-D trajectories of objects or of body parts and how to align similar moving objects but observed from different viewpoints with different cameras. The direct application of this technique is the synchronisation of two videos observing the same human gesture but gathered at different times and places.

In the future we plan to combine such methods with a fully articulated body part model in order to recover complex human motions from archived videos. Numerous applications in bio-mechanics, virtual reality, and so forth will then become available.

A Sensitivity to camera calibration errors

In this paper we assumed that the center of projection of the camera model lies at the image center. Obviously there is an error associated with this hypothesis and we derive an analytical expression allowing to appreciate the sensitivity of the final results with respect to this error. Let $\mathbf{H}$ be the homography estimated between images 1 and 2:

$$\mathbf{H} = \mathbf{K}_1\mathbf{R}\mathbf{K}_2^{-1} = \widehat{\mathbf{K}}_1\widehat{\mathbf{R}}\widehat{\mathbf{K}}_2^{-1}$$

with the following definitions:

$$\mathbf{K} = \begin{bmatrix} f & 0 & u \\ 0 & f & v \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \widehat{\mathbf{K}} = \begin{bmatrix} f & 0 & u+du \\ 0 & f & v+dv \\ 0 & 0 & 1 \end{bmatrix}$$

where f, u, and v are the optimal values for the internal parameters and du and dv are the errors committed when the principal point is set to be at the image center.

From the equation above we get:

$$\widehat{\mathbf{R}} = \widehat{\mathbf{K}}_1^{-1}\mathbf{K}_1\ \mathbf{R}\ \mathbf{K}_2^{-1}\widehat{\mathbf{K}}_2$$

Notice that from [10] we have $\widehat{\mathbf{K}}^{-1}\mathbf{K} = \mathbf{I} - \mathbf{E}$ and that its inverse is approximated by $\mathbf{I}+\mathbf{E}$ with:

$$\mathbf{E} = \begin{bmatrix} 0 & 0 & \frac{du}{f} \\ 0 & 0 & \frac{dv}{f} \\ 0 & 0 & 0 \end{bmatrix}$$

Within the context of frame-to-frame alignment, the rotation matrix is approximated by $\mathbf{R} = \mathbf{I} + [\mathbf{\Phi}]_\times$ (see section 2 and eq. 3) where $\mathbf{\Phi} = (\phi^x\ \phi^y\ \phi^z)$ are small angles. The estimated rotation becomes:

$$\begin{aligned} \widehat{\mathbf{R}} &= (\mathbf{I} - \mathbf{E})(\mathbf{I} + [\mathbf{\Phi}]_\times)(\mathbf{I} + \mathbf{E}) \\ &= \mathbf{I} + [\mathbf{\Phi}]_\times + \underbrace{[\mathbf{\Phi}]_\times\mathbf{E} - \mathbf{E}[\mathbf{\Phi}]_\times}_{\text{second-order}} + \underbrace{\mathbf{E}[\mathbf{\Phi}]_\times\mathbf{E}}_{\text{third-order}} \end{aligned}$$

Numerically, the average size of the focal length is, in this case, 800 pixels. Hence, if the absolute error in image center location is of about 40 pixels we get $du/f = 0.05$. Since the angles of rotation are small, the second- and third-order terms can be neglected. Therefore, a rough guess of the camera center of projection has little influence on the final alignment.

B Incremental computation of a homography

In this appendix we derive the expression of $\mathbf{H}_{\text{imp}}$ given by eq. (10). The current estimate of the homography, $\mathbf{H}_{\text{old}}$ writes:

$$\mathbf{H}_{\text{old}} = \mathbf{K}_2\mathbf{R}_{\text{old}}\mathbf{K}_1^{-1}$$

The new estimate of the homography, $\mathbf{H}_{\text{new}}$ differs from the current (old) one by an incremental improvement of its parameters:

$$\mathbf{H}_{\text{new}} = \mathbf{K}_2 \left(\mathbf{I} + \mathbf{K}_f\right) \ \mathbf{R}_\phi \mathbf{R}_{\text{old}} \ \mathbf{K}_1^{-1}$$

with:

$$\mathbf{K}_1 = \begin{bmatrix} f_1 & 0 & 0 \\ 0 & f_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{K}_2 = \begin{bmatrix} f_2 & 0 & 0 \\ 0 & f_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{K}_f = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Hence,$f_2(1+f)$ is the new focal length and $\mathbf{R}_\phi$ is a small rotation, $\mathbf{R}_\phi = \mathbf{I} - [\mathbf{\Phi}]_\times$.

By developing the expression above and grouping terms appropriately we obtain:

$$\begin{aligned} \mathbf{H}_{\text{new}} &= \left(\mathbf{I} + \mathbf{K}_f - \mathbf{K}_2[\mathbf{\Phi}]_\times \mathbf{K}_2^{-1}\right) \ \mathbf{K}_2 \mathbf{R}_{\text{old}} \mathbf{K}_1^{-1} \\ &= \left(\mathbf{I} + \mathbf{\Sigma}\right) \mathbf{H}_{\text{old}} \end{aligned}$$

Noticing that the inverse of $\mathbf{I} + \mathbf{\Sigma}$ is $\mathbf{I} - \mathbf{\Sigma}$ (this is because we deal with small angular and small focal length increments) we obtain for the improvement:

$$\mathbf{H}_{\text{imp}} = \mathbf{I} - \mathbf{K}_f + \mathbf{K}_2[\mathbf{\Phi}]_\times \mathbf{K}_2^{-1}$$

which corresponds to eq. (10).

References

[1] D. C. Alexander and B. F. Buxton. Statistical modeling of colour data. *International Journal of Computer Vision*, 44(2):87–109, September 2001.

[2] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 237–252, 1992.

[3] S.E. Chen. Quicktime VR - an image-based approach to virtual environment navigation. In SIGGRAPH *1995, Los Angeles, USA*, pages 29–38, 1995.

[4] N. Dalal and R. Horaud. Indexing key positions between multiple videos. In *Proceedings of IEEE Workshop on Motion and Video Computing*, pages 593–598, Orlando, Florida, USA, December 2002.

[5] DARTFISH Ltd. *DartTrainer Software and User's manual*, 2001/2002. http://www.dartfish.com/.

[6] F. Dufaux and F. Moscheni. Background mosaicking for low bit rate video coding. In *Proceedings IEEE International Conference on Image Processing*, volume 1, pages 673–676, Lausanne, Switzerland, September 1996.

[7] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, Cambridge, UK, 2000.

[9] R. I. Hartley. Self-calibration from multiple views with a rotating camera. In *Proc. Third European Conference on Computer Vision*, pages 471–478, Stockholm, Sweden, May 1994.

[10] R. Horaud, G. Csurka, and D. Demirdjian. Stereo calibration from rigid motions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1446–1452, December 2000.

[11] M. Irani and P. Anandan. About direct methods. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, number 1883 in LNCS, pages 267–277, Corfu, Greece, July 1999. Springer-Verlag.

[12] M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu. Efficient representations of video sequences and their applications. *Signal Processing: Image Communication, special issue on Image and Video Semantics: Processing, Analysis, and Application*, 8(4), May 1996.

[13] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 605–611, June 1995.

[14] J.M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348–365, December 1995.

[15] S. Peleg and J. Herman. Panoramic mosaics by manifold projection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Puerto Rico, USA*, pages 338–343, 1997.

[16] S. Peleg, B. Rousso, A. Rav-Acha, and A. Zomet. Mosaicing on adaptive manifolds. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1144–1154, October 2000.

[17] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C - The Art of Scientific Computing.* Cambridge University Press, 2nd edition, 1992.

[18] REALVIZ S.A. *Stitcher 3.5 Software and User's Manual*, 2002. http://www.realviz.com/.

[19] H.S. Sawhney, S. Ayer, and M. Gorkani. Model-based 2D & 3D dominant motion estimation for mosaicing and video representation. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 583–590, June 1995.

[20] H.-Y. Shum and R. Szeliski. Systems and experiment paper: Construction of panoramic mosaics with global and local alignment. *International Journal of Computer Vision*, 36(2):101–130, February 2000.

[21] P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1), 2000.

[22] P. H. S. Torr and A. Zisserman. Feature based methods for structure and motion estimation. In W. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, number 1883 in LNCS, pages 278–295, Corfu, Greece, July 1999. Springer-Verlag.

[23] B. Triggs, P.F. McLauchlan, R.I. Hartley, and A. Fitzgibbon. Bundle ajustment — a modern synthesis. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice, Corfu, Greece*, volume 1883 of *Lecture Notes in Computer Science*, pages 298–372. Springer-Verlag, 2000.

[24] J.Y.A. Wang, E.H. Adelson, and U. Desai. Applying mid-level vision techniques for video data compression and manipulation. In *Proceedings of the* SPIE*: Digital Video Compression on Personal Computers, San Jose*, volume 2187, February 1994.

[25] I. Zoghlami, O. Faugeras, and R. Deriche. Using geometric corners to build a 2D mosaic from a set of images. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Puerto Rico, USA*, pages 420–425, June 1997.

Figure 1: This figure shows 5 images extracted from a 350-frame sequence (top), the static panorama or the background image (middle) showing the static objects used to estimate the time-varying camera parameters (focal length, pan and tilt angles), as well as the motion panorama showing a high-jump athlete in various postures as it would have been filmed with a wide-angle static camera (bottom). Notice the fine image resolution associated with the output images.

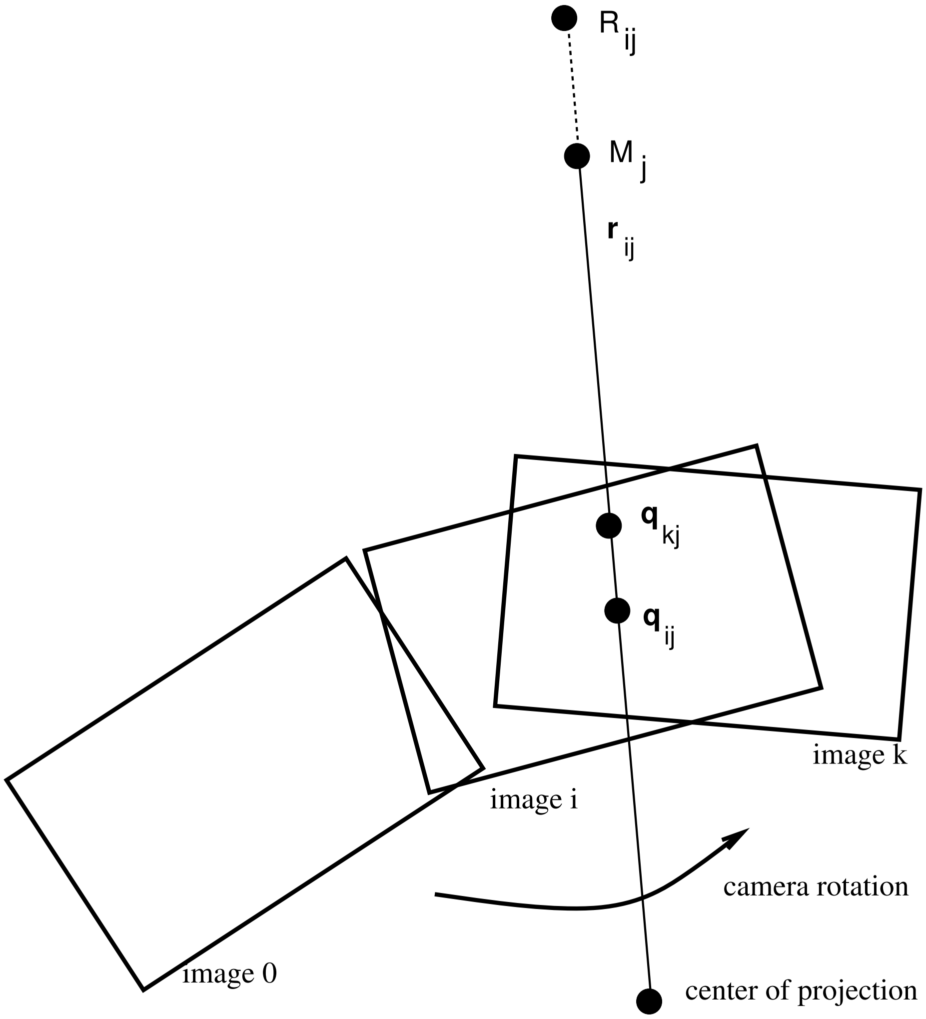

Figure 2: While the camera rotates from position “0” to “i” and then to “k”, point M_j is observed in image i and then in image k.

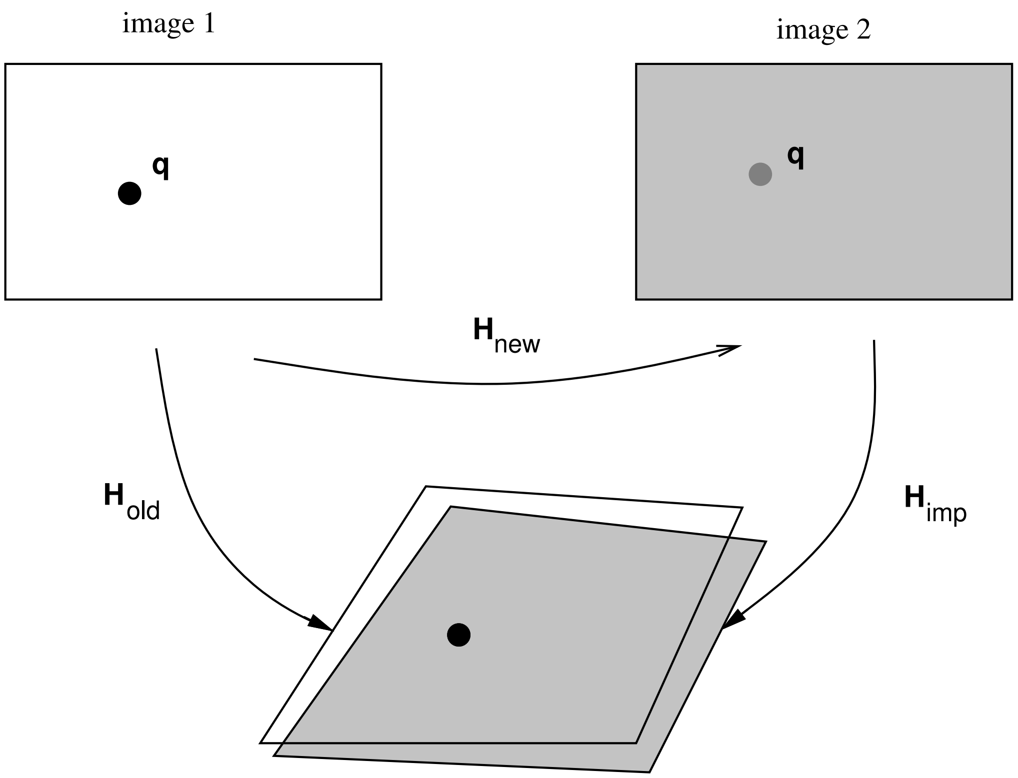

Figure 3: Incremental estimation of the homography between two images. This figure shows the factorisation of $\mathbf{H}_{\mathrm{old}}$ into two matrices, $\mathbf{H}_{\mathrm{imp}}$ and $\mathbf{H}_{\mathrm{new}}$. This factorisation allows a simpler and faster iterative estimation of the homography by the direct method.

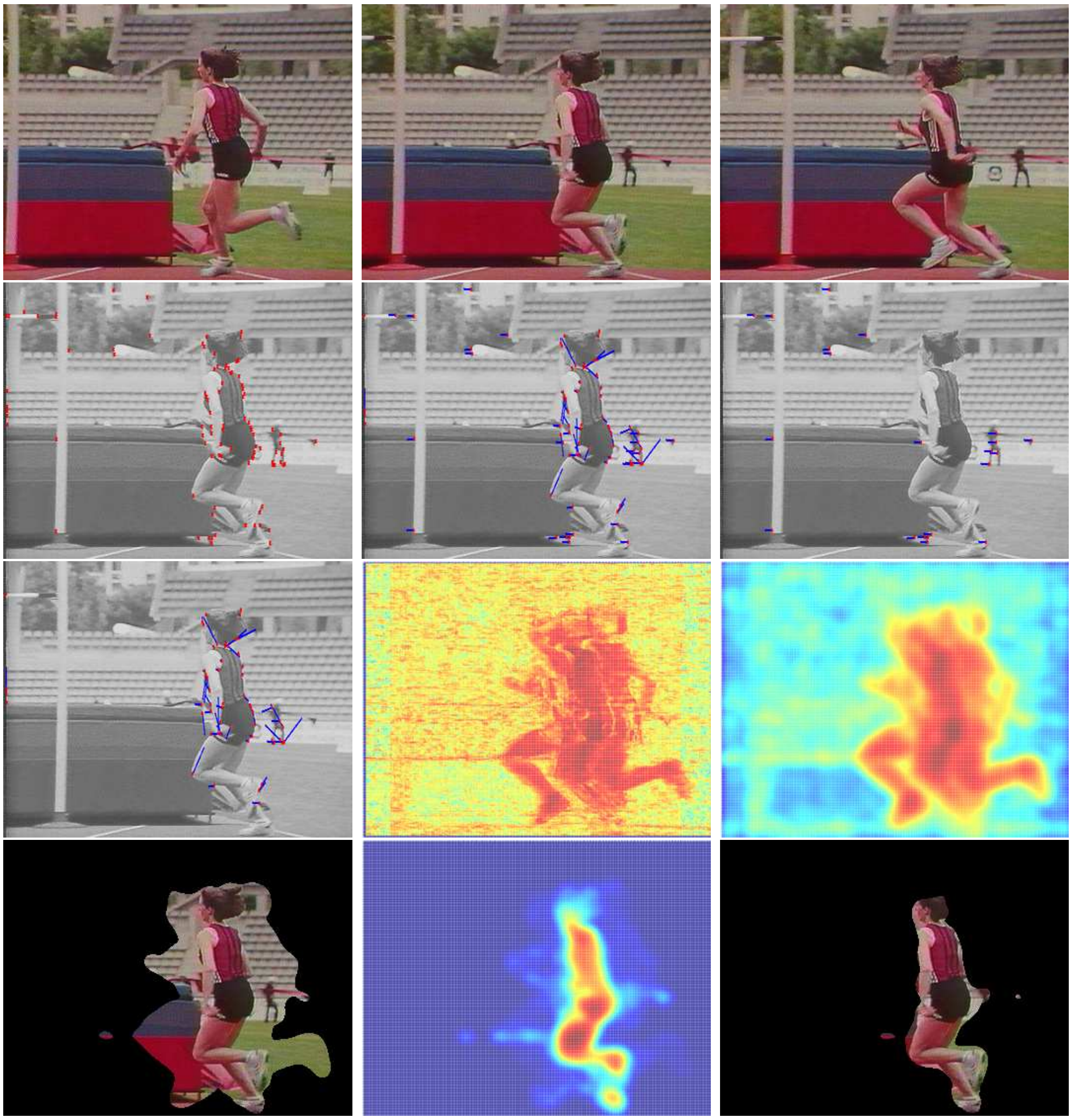

Figure 4: Left to right and top to bottom: The top line in this figure shows frames 218 (previous), 222 (current), and 226 (next). The results obtained with the current frame (222) are shown next. Interest points are first extracted and matched based on cross-correlation. This set of matched features is next divided into inliers and outliers. The inliers obey a rotational camera motion model with a constant focal length. The probability of foreground (red) is then obtained by comparing the three frames. This probability is then smoothed and the foreground is extracted. Once the direct method is applied to the remaining background, the probability of foreground is estimated again and the foreground region is finally extracted.

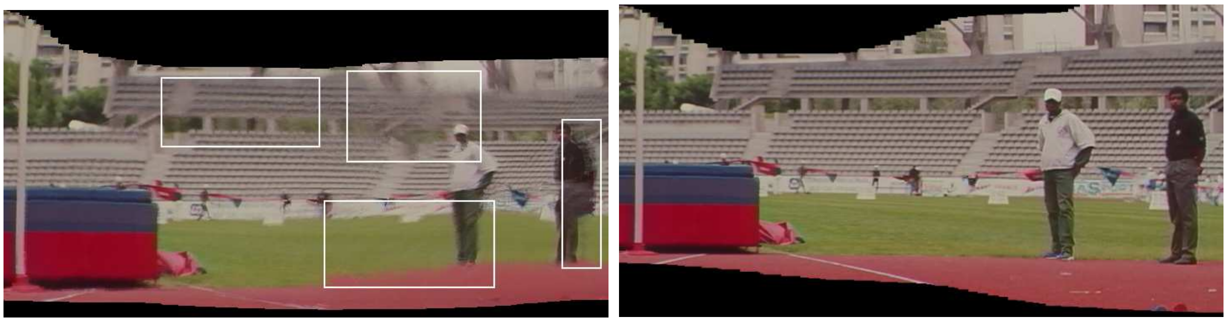

Figure 5: Detail of the background panorama aligned with the feature-based registration method (left) and the same panorama obtained by refining the alignment with the direct frame-to-frame registration method (right).

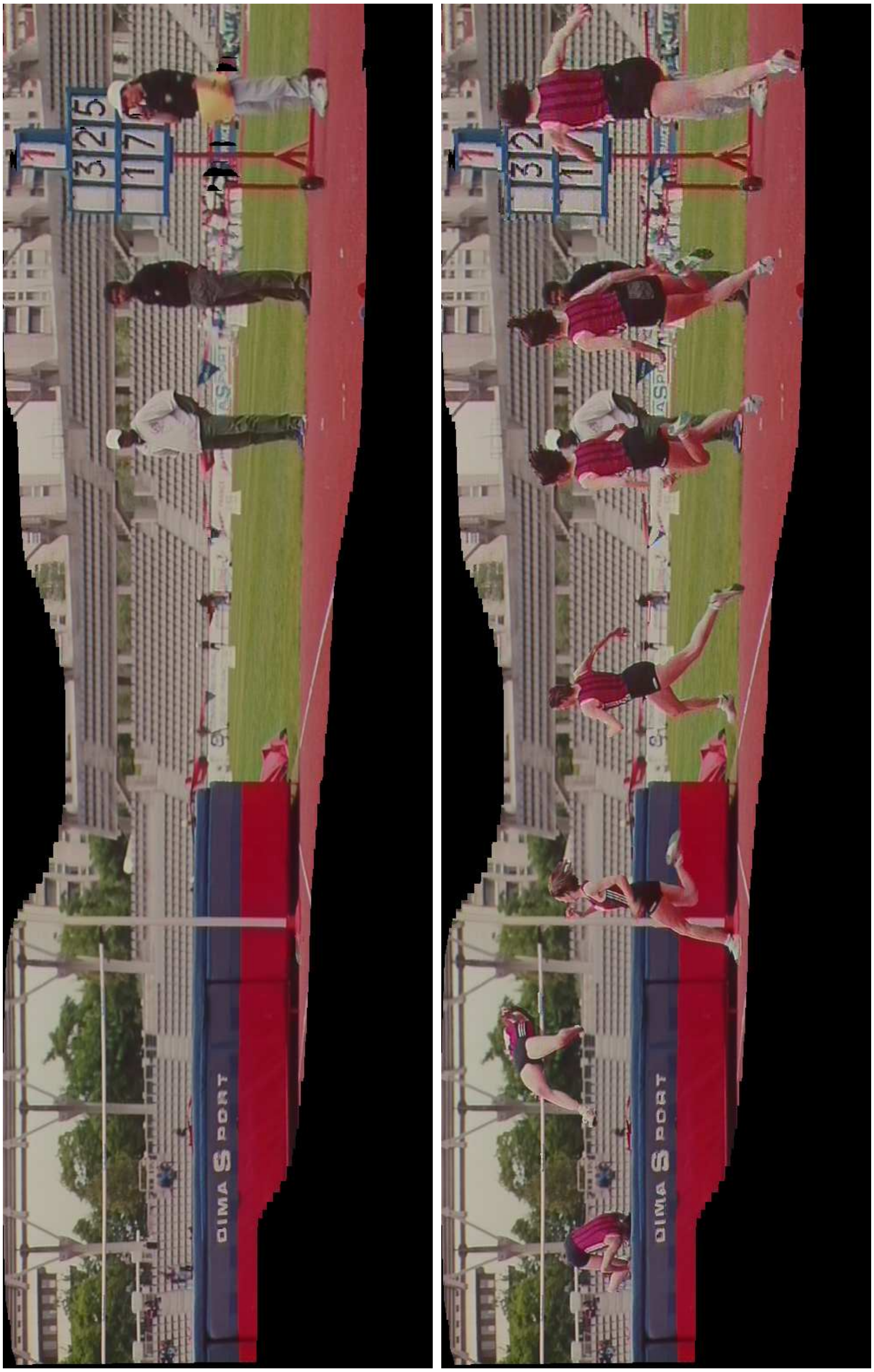

Figure 6: Background and motion panoramas for the high-jump video sequence.

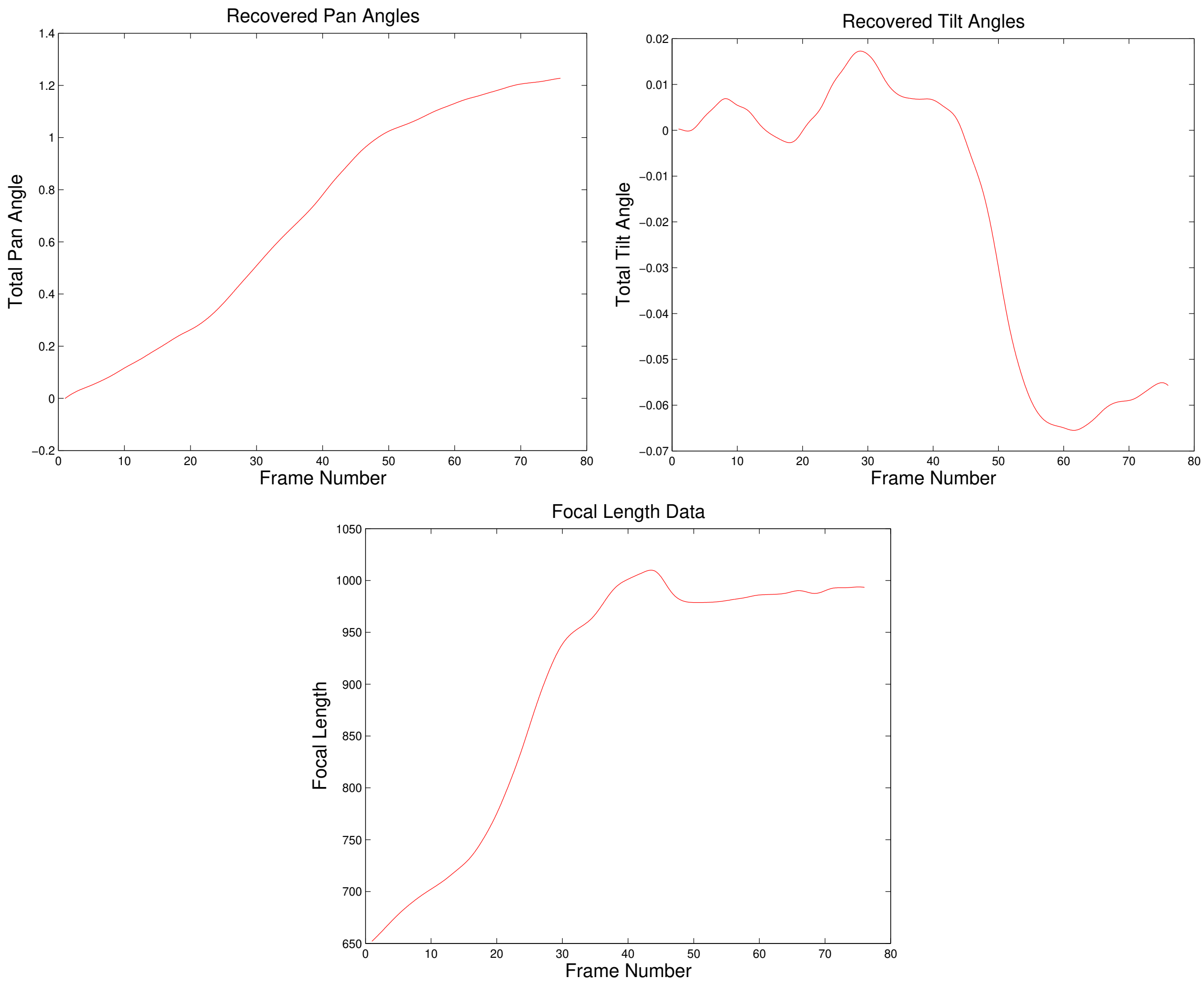

Figure 7: The pan (lateral camera rotation), tilt (up-and-down camera rotation) and zoom settings for the high-jump sequence.

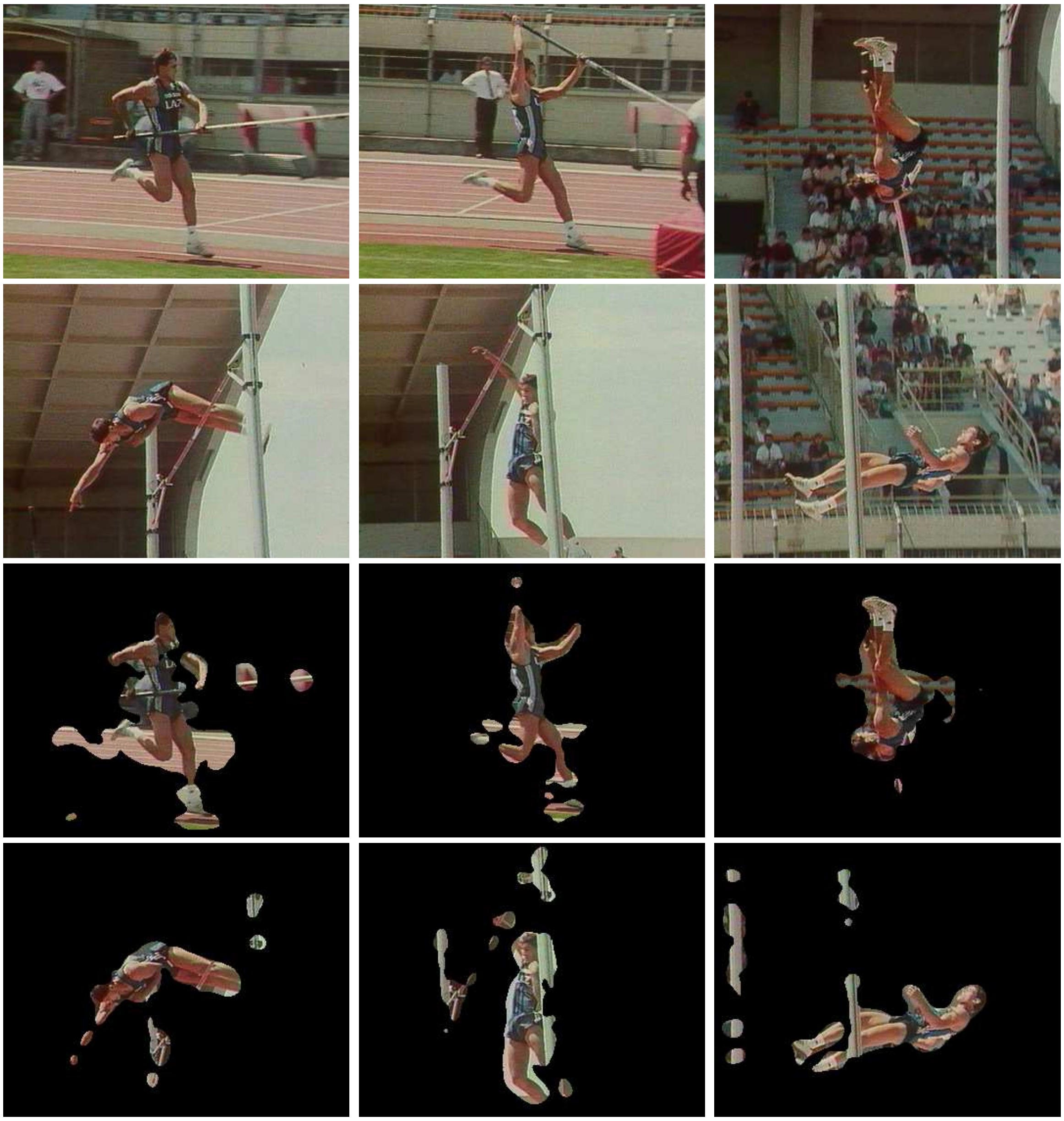

Figure 8: Sample frames (top) from a pole-vault video and the corresponding foreground regions (down).

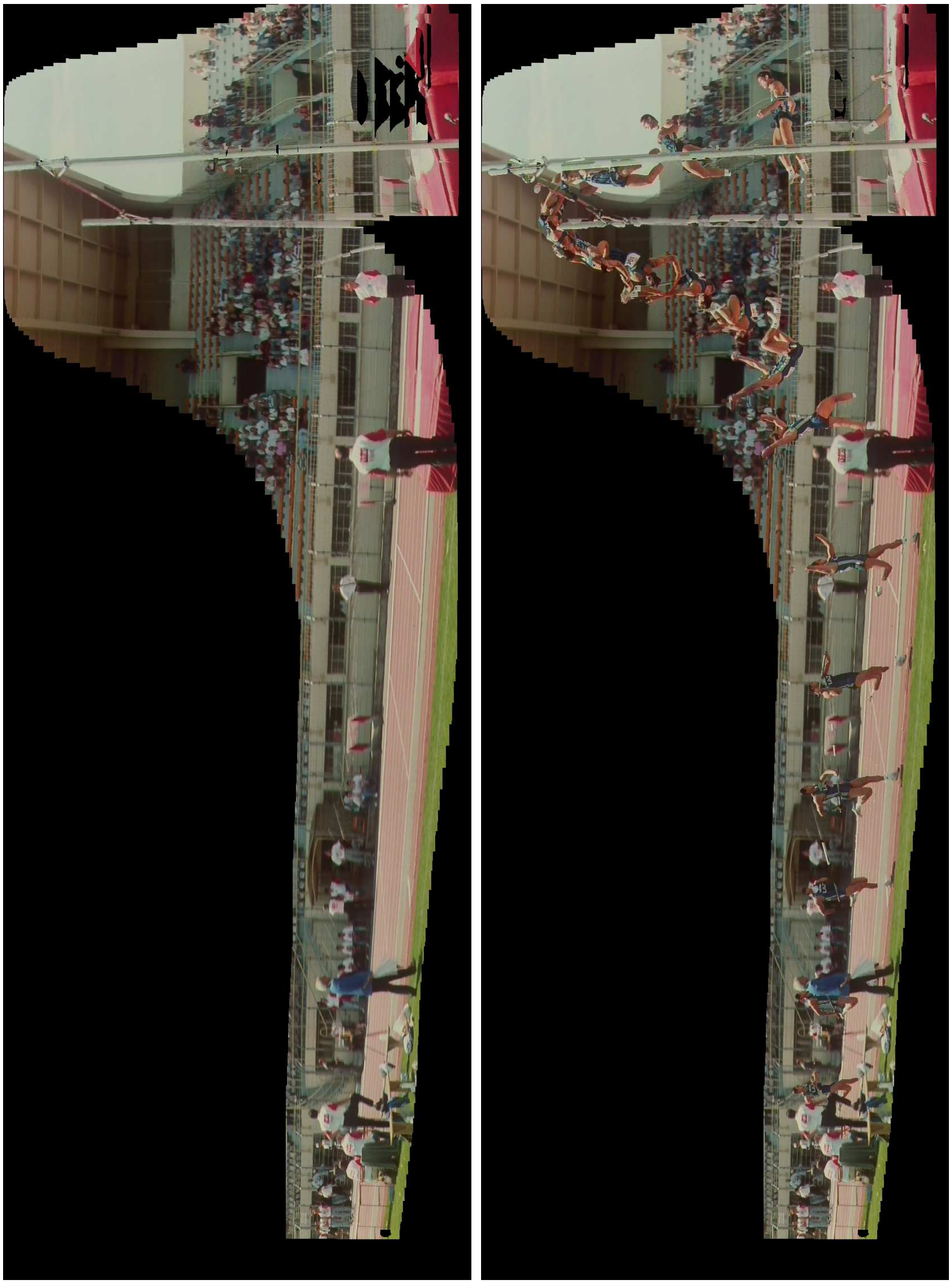

Figure 9: Background and motion panoramas for the pole-vault video sequence.