From Video Sequences to Motion Panoramas

Adrien Bartoli, Navneet Dalal, Biswajit Bose and Radu Horaud
INRIA Rhône-Alpes, 655, av. de l'Europe
38334 Saint Ismier cedex, France. *first.last@inria.fr*

Abstract

We address the problem of constructing mosaics from video sequences taken by rotating cameras. In particular, we investigate the widespread case where the scene is not only static but may also contain large dynamic areas, induced by moving or deforming objects. Most of the existing techniques fail to produce reliable results on such video sequences.

For such alignment purposes, two classes of techniques may be used: feature-based and direct methods. We derive both of them in a unified statistical manner and propose an integrated framework to construct what we call motion panoramas, based on a mixed feature-based and direct approach.

Experimental results are provided on large image sequences. In particular, we consider sport videos where the moving and deforming athlet is visible in every frame of the sequence, thereby making tricky the alignment task.

1. Introduction

Panoramic photography has received a growing interest since a decade [2, 3, 7, 8, 11, 13, 15]. It consists in stitching images to form wide-angle mosaics. Among their numerous applications, such techniques may be used to efficiently represent video sequences, in terms of compression, enhancement, vizualization etc, see [8].

A number of papers, see e.g. [11], concentrate on the static case, i.e. they deal with video sequences of static scenes. While high-quality results are obtained, this assumption prunes many real-life video sequences. Other works, see e.g. [8] concentrate on the analysis of video sequences of dynamic scenes, i.e. that may contain motion and undergo deformations. Most techniques are based on the observation that many real-life video sequences consist of two layers: a static background and a dynamic foreground. Single axis rotation is a very common and natural way to shoot scenes and often arises in many real-life video sequences, where the camera undergoes in particular a panning and tilting motion. We call *motion panoramas* the kind of mosaics representing such video sequences. Figure 1 illustrates this representation by showing a real-life video sequence and its associated motion panorama. Building a motion panorama consists therefore in producing the following results: a background panorama and the registration of the sequence, i.e. the camera motion and the dynamic layer.

The closest work to ours is [8] where the authors propose the use of direct methods to register dynamic scenes. We found that this approach works fine when the dynamic areas are small compared to the scene size.

The most important features of the techniques that may be used to construct such mosaics are the ability to handle large dynamic layers, and the accuracy in frame alignment. Two classes of techniques can be used: feature-based [13] and direct [7] methods. Each of them has specific advantages and drawbacks. In particular, we observed that feature-based methods are more robust, in terms of outling features (those who lie on e.g. the dynamic layer and therefore, who do not fulfill the motion model) while direct methods are more accurate, in terms of frame alignment, i.e. camera motion estimation.

These two techniques seem therefore to be complementary. Based on these observations, we propose a two-step technique for the construction of motion panoramas. We propose to use a feature-based method to initialize the registration, i.e. compute camera motion and layer segmentation, and a direct method to finely tune this registration. Our approach is statistically well-motivated, and several contributions to both feature-based and direct methods are given. In particular, we propose in §4, an efficient feature-based registration method, starting from robust frame-to-frame registration and ending with global bundle adjustment. Next, we propose in §5 an integrated framework for motion panorama construction based on direct registration of frames. We improve the existing frame-to-frame registration methods. The next two sections propose respectively a statistical derivation of feature-based and direct methods, needed to motivate our framework, and the motion model being used and its self-calibration. Experimental results are provided throughout the paper and concluding remarks are given in §6.

Notation and preliminaries. Everything is expressed in homogeneous coordinates, i.e. up to scale. We model the

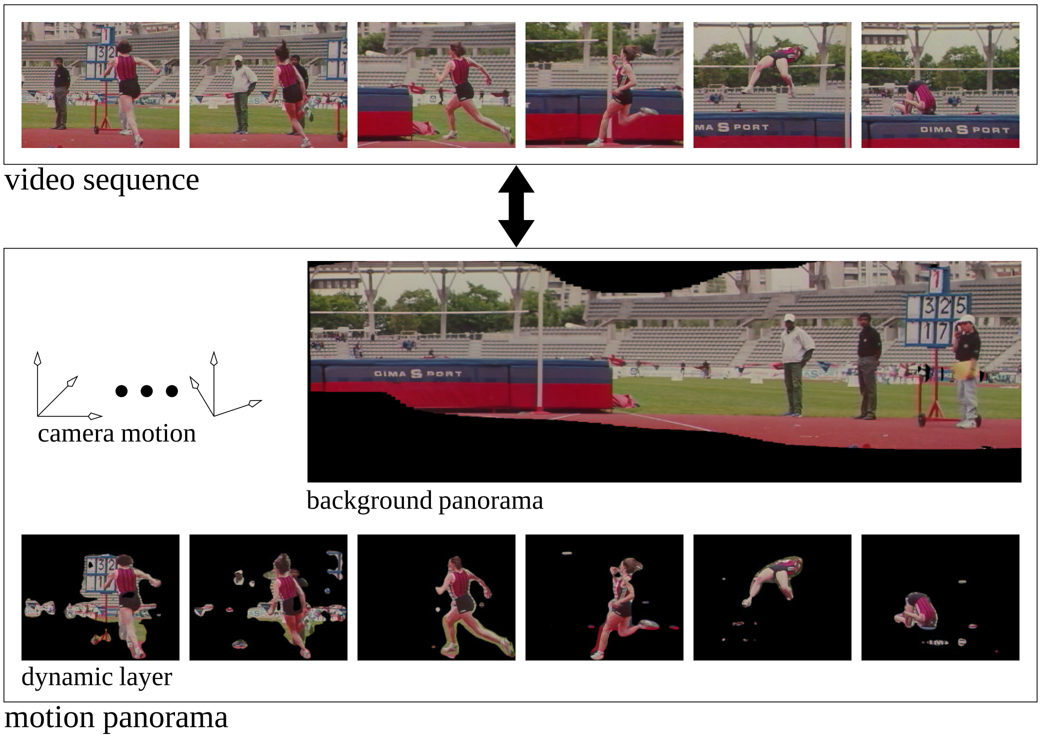


Figure 1. The 300-frame “high jump” video sequence and its motion panorama.

temporal aspect of entities by an index, usually i, designating the discrete time instant. We consider the pin-hole camera model, represented at each time instant i by a 3×4 perspective projection matrix P_i. We express all 3D entities in a coordinate frame centered on the position of the camera at the first time instant. In this case, if K_i is the intrinsic parameters of the camera and R_i its orientation, we can write $\mathsf{P}_i \sim (\ \mathsf{K}_i\mathsf{R}_i \ \ \mathbf{0})$, see e.g. [11].

Let $\mathbf{q}_i$ be an image point represented by a 3-vector. The corresponding 3D ray $\mathbf{r}$, i.e. the ray passing through the camera center and point $\mathbf{q}_i$, is obtained as $\mathbf{r} \sim (\mathsf{K}_i\mathsf{R}_i)^{-1}\mathbf{q}_i$.

From this, we can derive the inter-frame motion model, e.g. between frame i and frame j, for a point $\mathbf{q}$, as $\mathbf{q}_j \sim \mathsf{H}_{ij}\mathbf{q}_i$ where $\mathsf{H}_{ij} \sim \mathsf{K}_j\mathsf{R}_j\mathsf{R}_i{}^{-1}\mathsf{K}_i{}^{-1}$. We consider a sequence of m frames. Each frame i has an orientation R_i and intrinsic parameters K_i [11], denoted by $\mathcal{M}_i \equiv (\mathsf{R}_i, \mathsf{K}_i)$. 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 $\mathcal{I} \equiv \{\mathcal{I}_1, \ldots, \mathcal{I}_m\}$. Indices will sometimes be dropped for clarity.

2. Feature-Based and Direct Methods

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 maximizing the likelihood of the registration under the assumption of uniform probability on the registration and the scene.

Unfortunately, solving this problem is, in general, untractable. 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 class:

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

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 compute the registration which best explains these features:

$$\hat{\boldsymbol{\theta}} \approx \arg\max_{\boldsymbol{\theta}} \Pr(\mathcal{Q}|\boldsymbol{\theta}). \tag{2}$$

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 optimization techniques. The previous assumption of conditional independence for non-consecutive frames may be used to compute an initial guess for the registration.

The characteristics of feature-based and direct methods may be summarized as follows: feature-based methods can compute the maximum likelihood estimate of the registration over the complete sequence but only with respect to the features extracted from the images, while direct methods compute maximum likelihood estimate with respect to the images but only locally, i.e. from frame to frame. Moreover, the former may be highly robust while the latter may only tolerate few outliers, i.e. pixels lying on the dynamic layer.

3. The Motion Model

As said in the introduction, it is very likely that camera motion is not general. In particular, pure panning or tilting motions are often used. For example, the sequence shown on figure 1 consists of a large panning motion followed by a small tilt when the athlet jumps.

In this section, our aim is to determine which motion parameters can be computed in all these practical cases. This analysis allows to derive a specific motion model. Finally, we examine means to estimate the motion parameters, i.e. self-calibrate the camera.

Which motion parameters can be computed? It has been shown in [4, 6] that a pure panning or tilting motion, i.e. pure y- or x-axis rotation, allows to compute the motion parameters up to some assumptions. In particular, if one assumes zero skew, unit aspect ratio and principal point lying at the center of the image, then the orientation and the focal lengths can be estimated. However, in the case of a pure z-axis rotation, the focal lengths can not be estimated, even if they are constrained to be identical. Our unknown motion parameters are therefore, for each frame, its focal length and orientation.

The motion model. From this point, we consider that the effect of known intrinsics have been undone and consider therefore $\mathsf{K}_i \sim \mathrm{diag}(f_i, f_i, 1)$. The inter-frame motion is therefore given by:

$$\mathsf{H}_{ij} \sim \mathrm{diag}(f_j, f_j, 1)\ \mathsf{R}_{ij}\ \mathrm{diag}(1/f_i, 1/f_i, 1).$$

Self-calibration. We follow the approach proposed in [4], based on the dual image of the absolute conic, represented by the 3×3 matrix $\omega_i^\star \sim \mathsf{K}_i\mathsf{K}_i^\mathsf{T} \sim \mathrm{diag}(f_i^2, f_i^2, 1)$. The inter-frame motion H_{ij} provides constraints via $\mathsf{H}_{ij}\omega_i^\star\mathsf{H}_{ij}^\mathsf{T} \sim \omega_j^\star$. Extracting the focal lengths is straightforward.

4. Feature-Based Registration

Means for feature-based registration of the frames are proposed. For a review of these methods, see [13]. Point features are extracted using the Harris corner detector and putative frame to frame correspondences are computed. We show how to use the robust MLESAC algorithm [12] based on RANSAC [5], to estimate camera motion. In particular, we show how a two-point estimator may be efficiently drawn based on the assumption of pan and tilt motion. The resulting motion is used to bootstrap a maximum likelihood estimator which solves problem (2). Figure 2 (a) shows details of the background panorama obtained with the computed motion parameters. One notices that the alignment is correct, despite the large dynamic layer in the original images but that the inaccuracy results in several blurring artefacts.

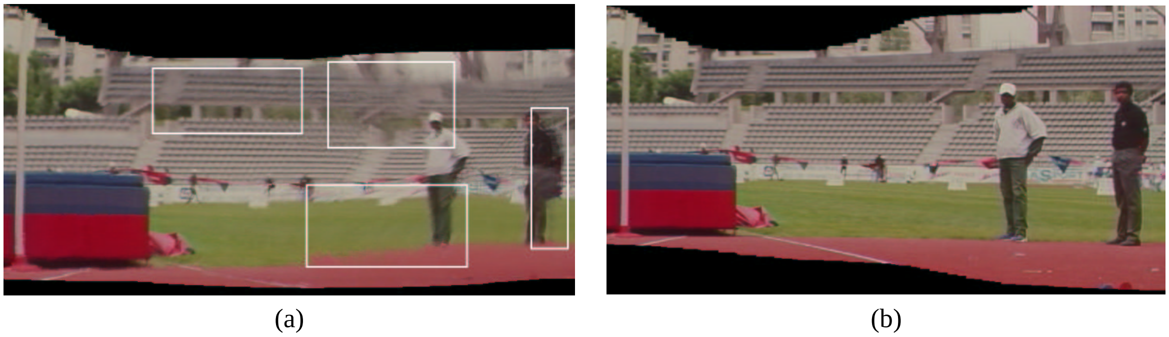
4.1. Initial Matching and Registration

We use a standard scheme for feature-based motion estimation [13]. We compute putative corner correspondences using correlation-based measures and then fit the motion model using MLESAC. MLESAC will provide us with a set of inliers, satisfying the dominant motion model, and a set of outliers. The set of outliers may be particularly wide, since each of them may correspond either to a spurious correspondence or to a point lying on the dynamic foreground.

Robust MLESAC registration. In brief, MLESAC consists in sampling a minimal set of correspondences, estimating the corresponding motion and computing its score, given by the robustified negative log-likelihood, related to the transfer error. This process is iterated a number of times that guarantees a probability of success, given a lower bound on the fraction of data contaminated by outliers. The highest-score motion parameters are selected. An efficient implementation is obtained by dynamically reducing the number of iterations, each time the estimated motion is improved, i.e. each time the lower bound on outliers is reduced. Once the algorithm has converged, we use the complete set of inliers to refine the motion estimation by minimizing the Euclidean distance between measured and transferred points. We use the Levenberg-Marquardt algorithm [10] for that purpose.

Obviously, the probability of success is guaranteed only if all computations are successful. Also, the less correspondences are necessary to compute motion, the less expensive will be the process. To this end, we need an estimator which uses the minimum number of correspondences and which can detect degenerate configurations. Such an estimator is proposed in the next paragraph.

A two-point motion estimator. If we consider that there is no rotation around the z-axis, then only four parameters need to be determined, namely the pan and tilt angles and the focal lengths. Therefore, two point matches should be



(a)

(b)

Figure 2. Detail of the background panorama constructed with the feature-based global maximum likelihood registration (a) and with its refinement using direct frame-to-frame registration (b).

enough to determine these parameters. Indeed, they give four linear constraints on the entries of $\mathsf{H} \sim \mathsf{K}'\mathsf{R}\mathsf{K}^{-1}$. However, a linear solution does not exist since the rotation component of H is not a linear function of the pan and tilt angles.

We propose to iteratively estimate the motion via local updates. For that purpose, we solve for f and f' while freezing R, then for R while freezing f and f'. Each of these subproblems is solved for linearly, by using a linear approximation of the rotation matrix in the second step. The process is iterated until convergence, assessed by thresholding the transfer error corresponding to the current motion estimate. Typically, 3 to 5 iterations are enough. The rotation is initialized to the identity. Degenerate configurations are detected by examining the rank of the matrix containing the four linear constraints given by the two point correspondences.

4.2. Maximum Likelihood Estimation

At this step, an initial estimate for camera motion, i.e. orientations and focal lengths is available. We now want to compute a global registration to reduce the effect of noise on these parameters. More precisely, we want to take into account point correspondences across more than two views, which allows to enforce constraints on the recovered parameters, such as constant focal length, and minimize a physically meaningful error.

Our estimator is inspired by bundle adjustment techniques, see e.g. [14] and solves problem (2). It consists in minimizing the reprojection error, defined as the discrepancy between measured and predicted features. The minimization is conducted over both the motion parameters and 3D rays. In the case of independent and identical Gaussian noise on feature positions, it is known to yield the maximum likelihood estimate. In [11], the authors propose to use such a technique to minimize the difference between 3D rays and not between reprojected features. While this technique sounds intuitively correct, it might induce a bias in the estimate, compared to the optimal technique. We use the Levenberg-Marquardt algorithm to conduct the optimization and compute the Jacobian matrix via finite differences [10], which we found to be as fast as the analytic differenciation proposed in [11]. An efficient implementation is obtained by considering the special sparse structure of the normal equations to be solved. Figure 3 shows the angles recovered by different methods. Note the instability of the feature-based frame-to-frame estimation. The motion recovered by bundle adjustment, i.e. global refinement, is much more stable. The severe discrepancy between the pan angles estimated by feature-based global bundle adjustment and direct refinement can be explained by the convergence of at least one of these methods to a local minima. One can observe that the peek in the tilt angle corresponds to the jump of the athlet while the peek in the pan angle corresponds to the highest speed of the athlet, just before the jump.

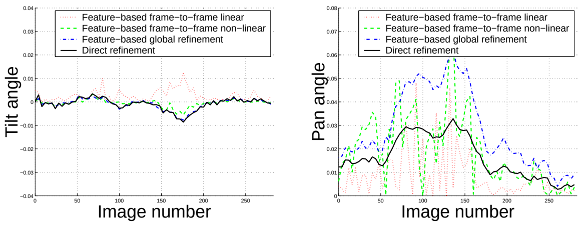


Figure 3. Recovered tilt and pan angles for the high jump sequence.

5. Construction of the Motion Panorama

We propose a means to construct the motion panorama, given the initial guess of camera motion previously estimated. We first compute the background panorama, then we extract the dynamic layer.

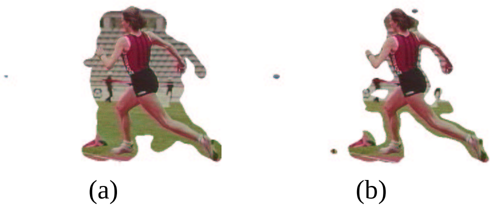


Figure 4. Dynamic layer for an input frame, extracted on a consecutive-frame basis (a) and with respect to the background panorama (b).

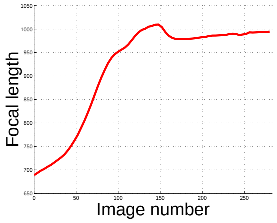


Figure 5. Recovered focal lengths for the high jump sequence via direct frame alignment.

5.1. Building the Background Panorama

We propose a three-step solution relying on a consecutive-frame (i.e. local in time) layer segmentation. We show how a subset of the background can be computed in each image. We use it to compute an initial guess of the background panorama. Based on this, we refine each frame registration using a direct method. For that purpose, we extend the registration method of [11], in particular, to allow for focal length refinement. Finally, we use this new registration parameters to finely tune the background panorama and the layer segmentation.

5.1.1 Initial Layer Segmentation

We propose to draw statistics on layers based on corresponding pixel colors in registered frames. Figure 4 (a) shows the dynamic layer extracted from an input frame. In order to avoid possible artefacts due to the presence of the dynamic layer, we propose to use only neighbouring frames. By doing so, we ensure that the fraction of the images considered which is affected by the dynamic layer is reduced. Obviously, the more images we use to draw statistics, the more stable the results will be, but only for pixels that never belong to the dynamic layer. The more images we use, the smaller is the area containing dynamic-layer-free information.

In more detail, to segment a given frame i, i.e. compute the dynamic layer $\mathcal{F}_i$, we propose to use the previous and next frames. Contrarily to [8], we do not need to guess the background color at this step. We only need to segment the image. We postpone the estimation of background color to the third step, see §5.1.3, when both the segmentation and the refined alignement of §5.1.2 will be available.

For each pixel $\mathbf{q}_i$ of the i-th frame, we wish to compute its probability to lie in the dynamic layer (i.e. the foreground) $\mathcal{F}_i$, knowing the images and the motion: $\mathrm{Pr}(\mathbf{q}_i \in \mathcal{F}_i | \mathcal{I}, \mathcal{M})$. Assuming conditional independence of non-consecutive frames, we may rewrite it as:

$$\mathrm{Pr}(\mathbf{q}_i \in \mathcal{F}_i | \mathcal{I}_{i-1}, \mathcal{I}_i, \mathcal{I}_{i+1}, \mathcal{M}_{i-1}, \mathcal{M}_i, \mathcal{M}_{i+1}),$$

and modeling the noise in image measurements by a Gaussian distribution, we obtain its log-likelihood p_i as:

$$p_i \propto d_{\mathsf{C}}(\mathcal{I}_{i-1}(\mathbf{q}_{i-1}), \mathcal{I}_i(\mathbf{q}_i)) + d_{\mathsf{C}}(\mathcal{I}_{i+1}(\mathbf{q}_{i+1}), \mathcal{I}_i(\mathbf{q}_i)), \quad (3)$$

where $d_{\mathsf{C}}(.,.)$ is the Mahalanobis distance defined by $d_{\mathsf{C}}(\mathbf{v}, \mathbf{v}') = \mathbf{v}^{\mathsf{T}} \mathsf{C}^{-1} \mathbf{v}'$ and C is the covariance matrix for the RGB color space, estimated using RGB color values from all pixels for each image in the video sequence. Under the general imaging conditions and without a prior color model, this tends to be one of the most reliable measures [1] of estimating distances in RGB color space.

We compensate for unmodeled deformations such as the radial distorsion induced by the camera by looking for the best corresponding pixel in the previous and next images around a neighbourhood of $\mathbf{q}_{i-1}$ and $\mathbf{q}_{i+1}$ respectively. This is equivalent to applying a shuffle transformation, previously used for multiple-view stereo in [9]. In practice, we use a one pixel radius transformation. Finally, we smooth the probability map obtained to remove high-frequency noise and threshold it using a χ^2 test to obtain the final dynamic map $\mathcal{F}_i$.

5.1.2 Refining Registration Using Direct Alignment

Up to now, we have a guess of the dynamic layer, which may therefore be pruned out of the estimation. Direct methods can then be applied directly on the remaining frame areas. Figure 2 (b) shows the background panorama obtained after direct alignment. One can observe that the inaccuracy of the initial feature-based method shown on figure 2 (a) is overcome while the convergence is ensured, due to the robust initialization.

We first present the algorithm of [2, 11]. We show that this framework has several drawbacks, namely, the focal lengths and the orientation can not be optimized at once. The motion update in the optimization loop is approximated to first order and at each loop, the gradient of an entire image needs to be computed, which may be expensive. We propose a method to avoid these drawbacks. Figure 5 shows the recovered varying focal length for the high jump sequence, with our method. Note that it roughly corresponds to the depth of the athlet since camera parameters are tuned so that it is maintained in the image and has the same size through the sequence.

The framework proposed in [2] consists of four parts: pyramid construction, motion estimation, image warping and coarse-to-fine refinement. The motion is linearized and iteratively updated by using the brighness constancy assumption.

This algorithm corresponds to minimizing the difference between the two images after alignment, which yields the maximum likelihood estimate with respect to the images, i.e. solves problem (1).

A major point is to perform the motion estimation between the first image and a warped second image, which allows to estimate only incremental deformations. [11] proposes to update either the orientation up to first order as $\mathsf{R} \leftarrow \mathsf{R}(\mathrm{I} + [\boldsymbol{\delta}]_\times)$ or the focal lengths, which they assumed to be identical, as $f \leftarrow f(1 + \delta_f)$. Updating both the orientation and the focal length leads to non-linear expression in the [2, 11] framework .

Let $\boldsymbol{\delta}$ be the motion update parameters, i.e. $\boldsymbol{\delta} = (\delta_{\theta_x} \ \delta_{\theta_y} \ \delta_{\theta_z})$ to update the rotation orientation, and H the current motion estimate. The framework of [2, 11] consists of the following steps. First, warp the second image $\mathcal{I}'$ to $\tilde{\mathcal{I}}' = \mathsf{H}^{-1}\mathcal{I}'$. Then, register this image with the first one (requires to compute the gradient of image $\tilde{\mathcal{I}}'$). Finally, update the motion parameters and loop over these three steps until convergence.

Alignment of the warped image $\tilde{\mathcal{I}}'$ and the first one $\mathcal{I}$ is conducted by directly minimizing the intensity error to determine the motion update parameters $\boldsymbol{\delta}$. In more detail, the energy function $E(\boldsymbol{\delta})$ is minimized via first order expansion of the warped image, as follows:

$$E(\boldsymbol{\delta}) = ||\tilde{\mathcal{I}}' - \mathcal{I}||^2 \approx \sum_{\mathbf{q} \in \mathcal{I}} (\mathbf{g}_{\mathbf{q}}^{\mathsf{T}} \mathsf{J}_{\mathbf{q}}^{\mathsf{T}} \boldsymbol{\delta} + e_{\mathbf{q}})^2,$$

where high order terms have been ignored after expansion of $\tilde{\mathcal{I}}'(\widetilde{\mathbf{q}}')$ using Taylor series. Let $\mathbf{q}^{\mathsf{T}} \sim (x \ y \ 1)$ be pixel coordinates. $e_{\mathbf{q}}$ is the image error given by $e_{\mathbf{q}} = \tilde{\mathcal{I}}'(\mathbf{q}) - \mathcal{I}(\mathbf{q})$, $\mathbf{g}_{\mathbf{q}}^{\mathsf{T}} = \nabla\tilde{\mathcal{I}}'(\mathbf{q})$ is the gradient of image $\tilde{\mathcal{I}}'$ at $\mathbf{q}$ and $\boldsymbol{\delta}^{\mathsf{T}} = (\delta_{\theta_x} \ \delta_{\theta_y} \ \delta_{\theta_z})$ and:

$$\mathsf{J}_{\mathbf{q}}^{\mathsf{T}} = \begin{pmatrix} -xy/f & f + x^2/f & -y \\ -f - y^2/f & xy/f & x \end{pmatrix} \tag{4}$$

is the Jacobian of the warped point $\widetilde{\mathbf{q}}'$ with respect to $\boldsymbol{\delta}$.

The first drawback of this method is that the rotation is approximated up to first order. Another drawback is the computational cost: the gradient of a whole image has to be computed at each loop. Finally, only the orientation or the focal length are computed but not both at once.

To avoid these drawbacks, we propose to update the motion without using a linear approximation for the rotation, via $\mathsf{R} \leftarrow \delta_{\mathsf{R}} \cdot \mathsf{R}$ where $\delta_{\mathsf{R}} = \mathsf{R}_x(\delta_{\theta_x}) \cdot \mathsf{R}_y(\delta_{\theta_y})$. Rotation around the z axis could be incorporated in a straightforward manner by post-multiplying by $\mathsf{R}_z(\delta_{\theta_z})$ (R_x, R_y and R_z are rotations around the x-, y- and z-axis respectively). We also include the focal length of the second image, therefore letting it vary between images as $\mathsf{K}' \leftarrow \mathsf{K}' + \delta_{\mathsf{K}'}$. From this, we derive a multiplicative update rule for H as:

$$\mathsf{H} \leftarrow \delta_{\mathsf{H}} \cdot \mathsf{H} \text{ where } \delta_{\mathsf{H}} = (\mathsf{K}' + \delta_{\mathsf{K}'})\delta_{\mathsf{R}}\mathsf{K}'^{-1}$$

It turns out that this update may be computed linearly.

To avoid computing the gradient image at each loop, we expand the first image and not the second one. Since the first image is invariant through the iterations, its gradient need to be computed only once.

Let us consider varying pan and tilt and second focal length, i.e. $\boldsymbol{\delta}^{\mathsf{T}} = (\delta_{\theta_x} \ \delta_{\theta_y} \ \delta_{f'})$. The corresponding Jacobian is given by:

$$\bar{\mathsf{J}}_{\mathbf{q}}^{\mathsf{T}} = \frac{\partial \delta_{\mathsf{H}} \mathbf{q}}{\partial \boldsymbol{\delta}}^{\mathsf{T}} = \begin{pmatrix} (\mathsf{J}_{\mathbf{q}}^{\mathsf{T}})_{2\times 2} & \begin{matrix} x/f' \\ y/f' \end{matrix} \end{pmatrix},$$

where $(\mathsf{J}_{\mathbf{q}}^{\mathsf{T}})_{2\times 2}$ is the leading part of the Jacobian matrix defined by equation (4), where the third column (corresponding to the variation on the z axis) has been dropped. The simple form obtained is due to the fact that the estimate is updated. The Jacobian is therefore evaluated at $\boldsymbol{\delta} = (0 \ 0 \ 0 \ 0)$. Also, contrarily to [11], the update step is performed without linear approximation of the rotation.

Figure 3 shows the orientation recovered with the direct method, compared to feature-based results. For the tilt angle, we observe that the bundle adjustment and the direct method perform as well, while for the pan angle, there are large discrepancies between them.

5.1.3 Pasting the Images

We now have a set of finely registered images and the dynamic layer segmentation. We may warp each frame onto either a reference one or onto a 2D surface, such as a cylinder or a sphere.

However, we still have the problem of combining the pixels from different images. A natural idea to guess the background color for a given pixel is the following. For each pixel of the panorama, each of its n instances is considered, to form a $3 \times n$ RGB color matrix V. [8] proposed several means to extract the background color from V, by assuming that the background color is dominant. They give techniques such as temporal averaging, median filtering, and different weighting stategies. We tried these approaches and found, as in [8], that they create ghosting artefacts, in particular, on the areas of the panorama including dynamic layer. In our case, the dynamic layer goes throughout the panorama, so we must find a better means to recover the background. We propose a comprehensive weighted scheme, with weights modeling both the local relative registration stability and global variance of colors at each pixel. This weighting is statistically motivated and extends the adhoc methods proposed in [8].

To compute the color of a pixel $\mathbf{q}$ of the panorama, corresponding pixel colors $\{\mathcal{I}_i(\mathbf{q}_i)\}$ in the images are used, provided $\mathbf{q}_i \notin \mathcal{F}_i$. We propose to weight each pixel contribution from a frame i according to a confidence measure c_i on the registration of this frame, drawn from the probability $\Pr(\mathcal{M}_i|\mathcal{I})$. Assuming conditional independence of the pixels within frame i and conditional independence of non-consecutive frames, we end up with a product over all pixels of image i: $\prod_{\mathbf{q}_i \notin \mathcal{F}_i} \Pr(\mathbf{q}_i|\mathcal{M})$. The corresponding log-likelihood l_i is obtained by summing the p_i of equation (3). Using such a confidence measure, we weight accordingly the pixels of the i-th frame by the robustified probability w_i defined as:

$$w_i = \begin{cases} 1 & \text{if } l_i \leq \mu \\ \exp\left(-\frac{(l_i-\mu)^2}{2\sigma^2}\right) & \text{otherwise,} \end{cases}$$

where μ and σ are the mean and variance of the l_i over all the images of the sequence. To further remove any ghosting due to false background regions in input image, w_i is augmented by multiplying it by the Gaussian-based probability density estimated over the distribution of colors at each pixel in the background mosaic [8].

5.2. Extracting the Dynamic Layer

Once a static background panorama has been created, we can warp each individual frame on it and perform background subtraction, using the statistical framework given in §5.1.1. Figure 4 (b) shows the result on an input image sequence. Note the significant improvement with respect to the initial segmentation.

6. Conclusion

We addressed the issue of constructing motion panoramas from image sequences taken by a rotating camera. A statistical derivation of feature-based and direct method classes shows that a mixed approach may solve the problem.

We proposed such an approach within an integrated framework that automatically computes a motion panorama from an image sequence. In more detail, the result is a background panorama, the camera motion (its time-varying orientation and focal length), and the dynamic layer. Such a representation may be very efficiently compressed and used to render the original images, where the dynamic layer may have been removed or changed.

We proposed several improvements to existing feature-based and direct methods, and in particular, a direct method to estimate both the orientation and the focal length of the camera.

We demonstrated the efficiency of our approach on large video sequences containing wide dynamic areas. Our system ends up with successful extraction of the dynamic layer and accurate frame alignment. We believe that combined feature-based and direct methods may solve efficiently such mosaicing problems.

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. Bergen, P. Anandan, K. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *ECCV*, pages 237–252, 1992.

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

[4] L. de Agapito, R. Hartley, and E. Hayman. Linear self-calibration of a rotating and zooming camera. In *CVPR, Fort Collins, Colorado, USA*, 1999.

[5] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381 – 395, June 1981.

[6] R. I. Hartley. Self-calibration of stationary cameras. *International Journal of Computer Vision*, 22(1):5–23, 1997.

[7] M. Irani and P. Anandan. About direct methods. In *Vision Algorithms: Theory and Practice*, July 1999.

[8] 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.

[9] K. Kutulakos. Approximate N-view stereo. In *Proceedings of the 6th European Conference on Computer Vision, Dublin, Ireland*, 2000.

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

[11] 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.

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

[13] P. H. S. Torr and A. Zisserman. Feature based methods for structure and motion estimation. In *Vision Algorithms: Theory and Practice*, July 1999.

[14] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle ajustment — a modern synthesis. In *Vision Algorithms: Theory and Practice*, July 1999.

[15] 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.