

# Video Mosaicing using a Mutual Information-based Motion Estimation Process

Amaury Dame<sup>1</sup>, Eric Marchand<sup>2</sup>

<sup>1</sup> CNRS, IRISA, INRIA Rennes    <sup>2</sup> Université de Rennes 1, IRISA, INRIA Rennes

**Abstract**— This paper proposes a method to achieve parametric motion estimation based on mutual information and its use in video mosaicing applications. Sum of Squared Differences (SSD) is widely considered in motion estimation. Here, we consider another metric, Mutual Information (MI), which is far less sensitive to changes in the lighting condition, to occlusions, and to a wide class of non-linear image transformation. Results under various complex conditions are presented.

## I. INTRODUCTION

Image mosaics are a collection of overlapping images. The goal of the mosaicing problem is to find the transformations that relate the different image coordinates. Once the transformation between all the images is known, an image of the whole scene can be constructed. This problem requires to find a warping function that maps the coordinates of one image into the coordinate system of another image. When considering a video, one has to warp each new image into the coordinate system of the very first image of the video [15] [3] [8] [6].

This is basically a motion estimation process. One can consider to estimate this motion using matched keypoints as in [3] or using SSD based motion estimation as in [8] [6]. The latter approach is very efficient when image sequences are considered, that is, when displacements between one frame to an other are small. Indeed, if the motion is small enough motion estimation can be achieved by an energy minimization process. Such approaches are related to the early work of Lucas and Kanade [9] that use images difference as energy [1].

These classical SSD-based approaches are based on the temporal luminance constancy hypothesis. However, it is well known that this hypothesis can be easily violated leading to important errors in the motion estimation process. Dealing with target tracking illumination variations can be taken into account for example using M-estimator [7] [12] or by the illumination changes as a surface that evolves over time [14].

To deal with occlusions, and illumination variations we propose to use the mutual information [13], [17] as the alignment function, that is, as we will see, robust to all this variations of appearance. Mutual information, especially studied in medical image registration, tolerates such changes. It is another metric well adapted and more robust for the tracking or motion estimation problems. Since its derivative form have been recently studied in many works [16] [11] [5] [4], its application for mosaicing has been considered here.

In the remainder of this paper, we recall the MI principle and present the MI-based motion estimation process that

allows to estimate the dominant motion in the image. Finally, results are given on various image sequence.

## II. MOTION ESTIMATION

We recall here the basic principle of the differential motion estimation methods that can be used for motion estimation. These methods are based on the optimization of a similarity measure  $f$ . The goal is to estimate the displacement  $\mathbf{p}$  of an image template  $I^*$  in a sequence of images  $I_0..I_t$ . In the mosaicing application  $I^* = I_0$ . In the case of a similarity function, the problem can be written as:

$$\hat{\mathbf{p}}_t = \arg \max_{\mathbf{p}} (f(I^*, w(I_t, \mathbf{p}))) . \quad (1)$$

where we search the displacement  $\hat{\mathbf{p}}_t$  that maximizes the similarity between the template  $I^*$  and the warped current image  $I_t$ .

To solve the maximization problem, the assumption made is that the global displacement between two consecutive frames is quite small. The previous estimated displacement  $\hat{\mathbf{p}}_{t-1}$  can therefore be used as first estimation of the current displacement to perform the optimization of  $f$  and incrementally reach the best estimation  $\hat{\mathbf{p}}_t$ .

Multiple solutions are possible to perform the iterative process and compute the update of the current displacement parameters. Indeed Baker and Matthews showed that two formulations were equivalent [1]. In our case we consider the inverse compositional formulation which considers that the update is modifying the reference image, so that  $\Delta\mathbf{p}$  is chosen to maximize:

$$\Delta\mathbf{p}_k = \arg \max_{\Delta\mathbf{p}} f(w(I^*, \Delta\mathbf{p}), w(I_t, \mathbf{p}_k)) . \quad (2)$$

In this case the current parameters will be update using:

$$w(w^{-1}(\mathbf{x}, \Delta\mathbf{p}_k), \mathbf{p}_k) \rightarrow w(\mathbf{x}, \mathbf{p}_{k+1}) . \quad (3)$$

In this formulation, since the update parameters is applied to the reference image, the derivatives with respect to the displacement parameters will classically be computed using the gradient of the reference image. Thus, these derivatives can be partially precomputed and the algorithm is far less time consuming.

One essential choice remains the one of the alignment function  $f$ . One natural solution is to choose the function  $f$  as the sum of the squared differences (SSD) of the pixel intensities between the reference image and the transformed current image:

$$\hat{\mathbf{p}}_t = \arg \min_{\mathbf{p}} \sum_{\mathbf{x} \in ROI} (I^*(\mathbf{x}) - I_t(w(\mathbf{x}, \mathbf{p})))^2 \quad (4)$$

where the summation is computed on each point  $\mathbf{x}$  of the reference template, that is, the region of interest (ROI, in our case this is the whole image) of the reference image. As its definition suggests this similarity function is very sensitive to occlusions and illumination variations. Many solutions have been proposed to make the SSD robust. M-estimators robustifies the least squared problem toward occlusions [7] [12] and illumination modeling can be used to handle illumination changes [7] [14].

### III. MUTUAL INFORMATION-BASED MOTION ESTIMATION

#### A. Mutual information

Rather than comparing intensities, mutual information is the quantity of information shared between two random variables. Mutual information of two random variable  $I$  and  $I^*$  is then given by the following equation:

$$MI(I, I^*) = h(I) + h(I^*) - h(I, I^*). \quad (5)$$

where the entropy  $h(I)$  is a measure of variability of a random variable  $I$  (signal, image...). If  $r$  are the possible values of  $I$  and  $p_I(r) = P(I = r)$  is the probability distribution function of  $r$ , then the Shannon entropy  $h(I)$  of a discrete variable  $I$  is given by the following expression:

$$h(I) = - \sum_r p_I(r) \log(p_I(r)). \quad (6)$$

The probability distribution function of the gray-level values is then simply given by the normalized histogram of the image  $I$ . The entropy can therefore be considered as a measure of dispersion of the image histogram.

Following the same principle, joint entropy  $h(I, I^*)$  of two random variables  $I$  and  $I^*$  can be defined as the variability of the couple of variables  $(I, I^*)$ . The Shannon joint entropy expression is given by:

$$h(I, I^*) = - \sum_{r,t} p_{II^*}(r, t) \log(p_{II^*}(r, t)) \quad (7)$$

where  $r$  and  $t$  are respectively the possible values of the variables  $I$  and  $I^*$ , and  $p_{II^*}(r, t) = P(I = r \cap I^* = t)$  is the joint probability distribution function. In our problem  $I$  and  $I^*$  are images. Then  $r$  and  $t$  are the gray-level values of the two images and the joint probability distribution function is a normalized bidimensional histogram of the two images. As for entropy, joint entropy corresponds to a measure of dispersion of the joint histogram of  $(I, I^*)$ .

The analytical formulation of a normalized histogram of an image  $I^*$  is classically written as follows:

$$p_I(r, \mathbf{p}) = \frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{x}} \phi(r - \bar{I}(w(\mathbf{x}, \mathbf{p}))) \quad (8)$$

$$p_{II^*}(r, t, \mathbf{p}) = \frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{x}} \phi(r - \bar{I}(w(\mathbf{x}, \mathbf{p}))) \phi(t - \bar{I}^*(\mathbf{x}))$$

where  $\mathbf{x}$  are the points of the region of interest in the image,  $N_{\mathbf{x}}$  is the number of points and  $t$  are the possible values of  $I^*(\mathbf{x})$ , i.e.  $t \in [0, N_c]$ . Let us note that to have a smooth mutual information it is important to maintain the low ( $N_c = 8$  in our implementation) the number of bins of the histogram (and thus to scale image intensity between

$\bar{I} \in [0, N_c]$ ). In the classical formulation  $\phi$  is a Kronecker's function:  $\phi(x) = 1$  for  $x = 0$  and  $\phi(x) = 0$  otherwise. So that each time  $I^*(\mathbf{x}) = i$  the  $i$ th histogram bin value is incremented. Nevertheless, several solutions have been proposed to simultaneously smooth the mutual information function and keep its accuracy [17] [10]. Our approach is based on the use of B-spline functions for  $\phi$  [10].

#### B. Mutual information-based motion estimation

In this section we will see how to use the MI cost function with the motion estimation process presented in section II. Let us remind that the goal is to estimate the displacement parameters  $\mathbf{p}_t$  that maximizes the MI using a first estimation of the parameters  $\mathbf{p}_{t-1}$  and an iterative update of the parameters.

a) *Derivative function analysis:* This problem implies a strong correlation between the elements of the vector  $\mathbf{p}$ . Therefore, the use of first-order optimization method such as the gradient "descent" is not adapted. Such non-linear optimization are usually performed using a Newton's method that assume the shape of the function to be parabolic.

Newton's method uses a second order Taylor expansion at the current position  $\mathbf{p}_{k-1}$  to estimate the update  $\Delta \mathbf{p}$  required to reach the optimum of the function (where the gradient of the function is null). The same estimation and update is performed until the parameter  $\mathbf{p}_k$  effectively reaches the optimum. The update is estimated following the equation:

$$\Delta \mathbf{p} = -\mathbf{H}^{-1} \mathbf{G}^{\top} \quad (9)$$

where  $\mathbf{G}$  and  $\mathbf{H}$  are respectively the Hessian and gradient matrices of the mutual information with respect to the update  $\Delta \mathbf{p}$ . Following the inverse compositional formulation defined in equation (2) those matrices are equal to:

$$\mathbf{G} = \frac{\partial MI(w(I^*, \Delta \mathbf{p}), w(I, \mathbf{p}))}{\partial \Delta \mathbf{p}} \quad (10)$$

$$\mathbf{H} = \frac{\partial^2 MI(w(I^*, \Delta \mathbf{p}), w(I, \mathbf{p}))}{\partial \Delta \mathbf{p}^2} \quad (11)$$

Applying the derivative chain rules yields the following gradient and Hessian matrices:

$$\mathbf{G} = \sum_{r,t} \frac{\partial p_{II^*}}{\partial \Delta \mathbf{p}} \left( 1 + \log \left( \frac{p_{II^*}}{p_I} \right) \right) \quad (12)$$

$$\mathbf{H} = \sum_{r,t} \frac{\partial p_{II^*}}{\partial \Delta \mathbf{p}} \frac{\partial p_{II^*}}{\partial \Delta \mathbf{p}} \left( \frac{1}{p_{II^*}} - \frac{1}{p_I} \right) + \frac{\partial^2 p_{II^*}}{\partial \Delta \mathbf{p}^2} \left( 1 + \log \frac{p_{II^*}}{p_I} \right) \quad (13)$$

For the purpose of clarity, the marginal probabilities and joint probability that are actually depending on  $r$ ,  $t$ ,  $\mathbf{p}^*$  and  $\Delta \mathbf{p}$  are simply denoted as  $p_I$ ,  $p_{I^*}$  and  $p_{II^*}$ . The details of the calculation from equation (10) to equation (12) can be found in [5] for a direct additional formulation.

By analogy with classical Hessian computation in SSD minimization, second order derivatives are usually neglected in the Hessian matrix computation [16] [5]. In our approach we compute the Hessian matrix using the second order

derivatives that are, in our point of view, required to obtain a precise estimation of the motion.

As we can see in equation (12) and equation (13), the derivatives of the mutual information depend on the derivatives of the joint probability. Using the previous definition in (8) and passing the derivative operator through the summation yields the following expressions:

$$\begin{aligned}\frac{\partial p_{II^*}}{\partial \Delta \mathbf{p}} &= \frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{x}} \phi(t - \bar{I}(w(\mathbf{x}, \mathbf{p}))) \frac{\partial \phi}{\partial \Delta \mathbf{p}}(r - \bar{I}^*(w(\mathbf{x}, \Delta \mathbf{p}))) \\ \frac{\partial^2 p_{II^*}}{\partial \Delta \mathbf{p}^2} &= \frac{1}{N_{\mathbf{x}}} \sum_{\mathbf{x}} \phi(t - \bar{I}(w(\mathbf{x}, \mathbf{p}))) \frac{\partial^2 \phi}{\partial \Delta \mathbf{p}^2}(r - \bar{I}^*(w(\mathbf{x}, \Delta \mathbf{p})))\end{aligned}$$

The remaining expressions to evaluate are the variations of the B-spline function  $\phi$  with respect to the update. Its derivatives are obtained using the chain rule and gives:

$$\begin{aligned}\frac{\partial \phi}{\partial \Delta \mathbf{p}}(r - \bar{I}^*(w(\mathbf{x}, \Delta \mathbf{p}))) &= -\frac{\partial \phi}{\partial r} \frac{\partial \bar{I}^*}{\partial \Delta \mathbf{p}} \\ \frac{\partial^2 \phi}{\partial \Delta \mathbf{p}^2}(r - \bar{I}^*(w(\mathbf{x}, \Delta \mathbf{p}))) &= \frac{\partial^2 \phi}{\partial t^2} \frac{\partial \bar{I}^*}{\partial \Delta \mathbf{p}} \frac{\partial \bar{I}^*}{\partial \Delta \mathbf{p}} - \frac{\partial \phi}{\partial r} \frac{\partial^2 \bar{I}^*}{\partial \Delta \mathbf{p}^2}.\end{aligned}\quad (14)$$

Finally the derivatives of the reference image intensity with respect to the update parameters  $\Delta \mathbf{p}$  is given by the following expressions:

$$\begin{aligned}\frac{\partial \bar{I}^*}{\partial \Delta \mathbf{p}} &= \nabla \bar{I}^* \frac{\partial w(\mathbf{x}, \mathbf{p})}{\partial \Delta \mathbf{p}} \\ \frac{\partial^2 \bar{I}^*}{\partial \Delta \mathbf{p}^2} &= \frac{\partial w}{\partial \Delta \mathbf{p}} \nabla^2 \bar{I}^* \frac{\partial w}{\partial \Delta \mathbf{p}} + \nabla \bar{I}^* \frac{\partial^2 w_x}{\partial \Delta \mathbf{p}^2} + \nabla \bar{I}^* \frac{\partial^2 w_y}{\partial \Delta \mathbf{p}^2}\end{aligned}\quad (15)$$

The motivation for using the inverse compositional formulation is then obvious. The derivatives of the warp function are all computed at  $\Delta \mathbf{p} = 0$ , their values are then constant for each pixels of the template. Moreover, since the reference image is constant, all the expressions from equation (14) to equation (16) are constants and have to be precomputed only one time.

To improve the optimization process we propose to use an approximation of the Hessian matrix computed in the case of a perfect alignment between the template and the current image that is given for  $\bar{I}(w(\mathbf{x}, \mathbf{p})) = \bar{I}^*(\mathbf{x})$  [4].

This solution has several advantages: first it gives a definite negative Hessian matrix that yields to have a wide convergence domain; second, since the Hessian matrix used in the Newton's method is the Hessian matrix after convergence, the behavior of the optimization near convergence is optimal and the final estimated displacement parameters is very accurate; finally, the Hessian matrix, representing 80 of the computation time in one iteration, is computed only one time.

In this work we focus on mosaicing application. Usually motion is mainly a rotation and a homography is then well suited. The warp function is thus defined by the group action  $w : \mathbb{SL}(3) \times \mathbb{P}^2$  with  $\mathbf{x} \in \mathbb{P}^2$  and  $\mathbf{p}$  defines the 8 parameters of the  $\mathfrak{sl}(3)$  lie algebra associated to the  $\mathbb{SL}(3)$  group. However, affin transformation has also been considered. All details regarding the derivatives of the chosen warp function can be found in [2].

## IV. EXPERIMENTAL RESULTS

These experiments show the application of the MI -based motion estimation algorithm to the mosaicing problem. In these sequence, since some part of scene completely disappears, it is necessary to define multiple reference images. The approach is build as follows:

- Initialization: the first image is chosen as reference image, i.e.  $I_0^* = I_0$ .
- Tracking: for every frame, we compute the displacement  $\mathbf{p}_k$  between  $I_t$  and  $I_k^*$ .
- Reference Update: every 30 images, the reference image  $I_k^*$  is changed and defined as the current image, i.e.  $I_k^* = I_t$  for  $t = 30k$ .

Using the homography from the current image to the current reference image and the homographies between the references, we retrieve the homography between the current image and the first image. Using this homography, we can project all the images of the sequence into the mosaic image and construct the global image of the whole scene.

In the first experiment (Figure 1 and 2), the overlapping images are simply a compressed sequence of 3600 images obtained from Youtube. The aerial scene is acquired from a camera embedded on a flying UAV and shows the ground that is approximately 1 kilometer away from the camera. Since this distance is very large, the scene can be approximated as a plane and tracked using homographies. During the acquisition of the sequence, the camera is moving forward and is rotating around the vertical axis.

In Figure 1 we show some images from the sequence. This sequence has been downloaded on Youtube and is affected by the H264 coding artifacts. We can also note the poor quality of the images. As we can see in the resulting mosaic image. Despite this poor quality, the resulting mosaic presented in Figure 2 shows the accuracy of the MI tracker. Since the camera is making an entire revolution, the first and last images are overlapping. A small shift occurs between the first and last estimated positions: we highlight two corresponding patterns that should have been at the same location on the mosaic. Let us note that nothing has been performed to reduce the drift (such as the bundle adjustment approach proposed by [3]). Considering the template update problem and the planar assumption, the estimated homographies are accurate. The same experiment was performed using the SSD tracker. In this case, due to the noise, blur, and illumination variations, this registration approach diverges after a few iterations.

The second experiment presents a mosaic build from more than 10000 images. Image are extracted from a highly compressed video. The camera was attached to a free flying balloon flying over Paris. Figure 3 shows three steps of the mosaic construction.

In this last experiment (see mosaic in Figure 4), we consider a sequence extracted from the John Ford movie “*she wore a yellow ribbon*”. In that case an affine motion model was consider. The interest of this sequence is that some cavalrymen are moving all along the sequence and, therefore, act as important occlusions as can be seen on Figure 5.



Fig. 4. Mosaic created from the John Ford movie “she wore a yellow ribbon”. A affine motion model was considered. Note that some cavalymen are moving all along the sequence. Despite these disturbances, motion is correctly estimated.

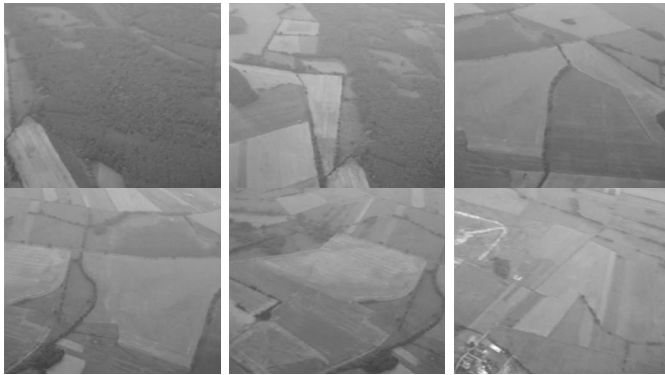


Fig. 1. Some overlapping key images used for the mosaicing application.

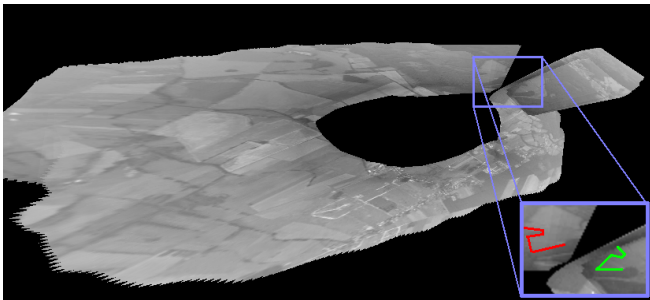


Fig. 2. Resulting mosaic image: despite the poor quality of the sequence and the approximation that the scene is planar, the final displacement between the first and last image is accurate. The red and green contours show the position of one physical pattern in the first and last images of the sequence.

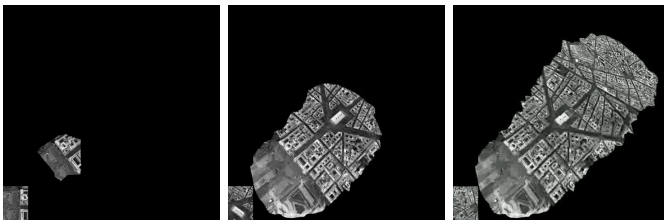


Fig. 3. Three steps of the “Paris” mosaic construction. The sequence feature more than 10000 images acquired from a camera attached to free-flying balloon.



Fig. 5. Three images used for the “yellow ribbon” mosaic construction.

## REFERENCES

- [1] S. Baker and I. Matthews. Equivalence and efficiency of image alignment algorithms. In *CVPR'01*, pp. 1090 – 1097, December 2001.
- [2] S. Benhimane and E. Malis. Homography-based 2d visual tracking and servoing. *Int. J. of Robotics Research*, 26(7):661–676, July 2007.
- [3] M. Brown and D.G. Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 74(1):59–73, 2007.
- [4] A. Dame and E. Marchand. Accurate real-time tracking using mutual information. In *IEEE ISMAR'10*, Seoul, Korea, October 2010.
- [5] N.D.H. Dowson and R. Bowden. A unifying framework for mutual information methods for use in non-linear optimisation. In *ECCV'06*, volume 1, pp. 365–378, June 2006.
- [6] N. Gracias and J. Santos-Victor. Underwater video mosaics as visual navigation maps. *CVIU*, 79(1):66 – 91, 2000.
- [7] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. on PAMI*, 20(10):1025–1039, October 1998.
- [8] M. Irani, P. Anandan, and S. Hsu. Mosaic based representations of video sequences and their applications. In *ICCV*, 1995.
- [9] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI'81*, pp. 674–679, 1981.
- [10] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE trans. on Medical Imaging*, 16(2):187–198, 1997.
- [11] F. Maes, D. Vandermeulen, and P. Suetens. Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information. *Medical Image Analysis*, 3(4):373–386, 1999.
- [12] J.-M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Com. and Image Representation*, 6(4):348–365, December 1995.
- [13] C.E. Shannon. A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.*, 5(1):3–55, January 2001.
- [14] G. Silveira and E. Malis. Real-time visual tracking under arbitrary illumination changes. In *CVPR'07*, Minneapolis, USA, June 2007.
- [15] R. Szeliski. Image alignment and stitching: a tutorial. *Found. Trends. Comput. Graph. Vis.*, 2:1–104, January 2006.
- [16] P. Thévenaz and M. Unser. Optimization of Mutual Information for Multiresolution Image Registration. *IEEE trans. on Image Processing*, 9(12):2083–2099, 2000.
- [17] P. Viola and W. Wells. Alignment by maximization of mutual information. *IJCV*, 24(2):137–154, 1997.