Multi-step flow fusion: towards accurate and dense correspondences in long video shots

Tomas Crivelli, Pierre-Henri Conze, Philippe Robert, Matthieu Fradet, Patrick Pérez

Multi-step flow fusion: towards accurate and dense correspondences in long video shots

Tomás Crivelli, Pierre-Henri Conze, Technicolor
Philippe Robert, Matthieu Fradet,
Patrick Pérez
firstname.lastname@technicolor.com

Abstract

The aim of this work is to estimate dense displacement fields over long video shots. Put in sequence they are useful for representing point trajectories but also for propagating (pulling) information from a reference frame to the rest of the video. Highly elaborated optical flow estimation algorithms are at hand, and they were applied before for dense point tracking by simple accumulation, however with unavoidable position drift. On the other hand, direct long-term point matching is more robust to such deviations, but it is very sensitive to ambiguous correspondences. Why not combining the benefits of both approaches? Following this idea, we develop a *multi-step flow fusion* method that optimally generates dense long-term displacement fields by first merging several candidate estimated paths and then filtering the tracks in the spatio-temporal domain. Our approach permits to handle small and large displacements with improved accuracy and it is able to recover a trajectory after temporary occlusions. Especially useful for video editing applications, we attack the problem of graphic element insertion and video volume segmentation, together with a number of quantitative comparisons on ground-truth data with state-of-the-art approaches.

1 Introduction

Tracking image points associated with scene fragments in video sequences is an important problem in computer vision. It indeed serves as a fundamental brick for a number of more advanced tasks such as structure-from-motion and camera tracking (e.g., for mobile robotics, scene reconstruction or augmented reality), visual tracking of objects (e.g., for visual servoing, surveillance, annotation or editing) and, more recently, video indexing (e.g., for video copy detection or video synchronization), action description, detection and recognition, and dynamic scene analysis at large.

In most cases, a sparse set of points, up to a few hundreds, is sufficient; these points are tracked independently based on their distinctive appearance. The standard tools for this type of point tracking are the KLT tracker [23] and its variants. There are cases though where *dense* sets of trajectories are better suited, if not mandatory. These include recent scene segmentation [4, 17] and analysis techniques [26] and a number of automatic and semi-automatic video editing tasks (e.g., graphic elements insertion [20] and 2D-to-3D video conversion [7]) in which spatial density and long-term temporal consistency are key.

For the purpose of tracking all points in a video, trackers of individual key points are not suited and resorting to a collective tracking is necessary. Considered over the shortest possible time interval in a given sequence, this problem basically amounts to estimating a dense displacement field, or *optical flow field*, between two consecutive video frames.

Optical flow techniques [5, 12, 19, 24, 28] thus appear as the natural tool to build dense point trajectories. Numerically, this amounts to temporal integration, for which classic tools such as Euler's and Runge-Kutta schemes are available. This is done for instance in [8, 17, 25]. Results are reported on fairly short sequences (reliable tracks last no more than about thirty frames). Unfortunately, no matter how good is the optical flow estimator, this leads to unavoidable error accumulation that results in a substantial drift over extended periods of time.

Sand and Teller [22] introduced *particle video*, a sophisticated framework to compute variable-length point trajectories from successive optical flows. To this end, they proposed a full-fledged optimization framework, combining standard dense motion accumulation with sparse feature matching tools, for a limited set of "particles" maintained under tracking and automatically selected on the image grid. There is a careful reasoning on occlusion and trajectory termination, however no accounting for temporarily occluded particles. In contrast, we strive for accuracy without renouncing to full tracking density and maximum possible track life-span. This will be clear in our quantitative experiments.

Garg *et al.* [11] recently proposed a method for computing optical flow between each of the images in a sequence and a reference frame adapted to *non-rigid surfaces*. They make use of the high correlation between 2D trajectories of different points and assume that the displacement of any point can be expressed compactly as a linear combination of a low-rank motion basis. This basis can be computed by applying Principal Component Analysis to a small subset of reliable point tracks. In contrast, our approach does not require prior feature tracking and strong *a priori* assumptions on scene contents.

If we focus only on the problem of colour video signal representation/reconstruction from a reference image, impressive dedicated algorithms were proposed in the literature (e.g., SIFT-Flow [18], PatchMatch [2], Coherency Sensitive Hashing [14]), not based on optical flow but establishing dense patch/feature correspondences. With a different aim, the physical interpretation of such correspondences as motion vectors is generally not possible given that images can share content only partly and loosely, with dramatic viewpoint and appearance differences. If we are to design a fairly general key-frame-based video editing tool, these approaches are not well suited.

With high quality editing of video shots of arbitrary duration in mind, we focus on the following problem: how to construct dense fields of correspondences over extended time periods using series of elementary optical flows. Toward this goal, the concept of *multi-step flow* was recently introduced in [9] to estimate, with high accuracy, *dense displacement fields* from any frame in a video to a common reference one, using sequences of optical flows. The idea is to rely on a set of displacement fields between arbitrarily distant frames. We develop significant improvements to this first approach based on three main extensions: 1) we extend the construction of candidate displacement fields by combination of bidirectional (forward and backward) elementary optical flows, 2) we formulate a sounder criterion for fusing flow field candidates, and 3) we develop a novel spatio-temporal filtering method exploiting trajectory-based features for refinement of long-term correspondence fields. The proposed *multi-step fusion* flow estimation technique performs well both for point-wise tracking and for "pulling" dense information from a reference frame, as demonstrated through a number of experiments on ground-truth data and through visual assessment.

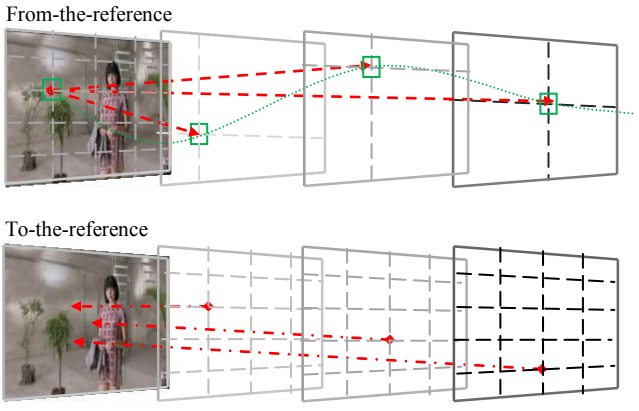


Figure 1. Point correspondence schemes. Top: *From-the-reference* scheme corresponds to the problem of determining the trajectory of each initial grid point in the reference frame along the sequence. Bottom: *To-the-reference* scheme corresponds to determining the position in the reference image of each grid point of each image of the sequence.

2 Multi-step flow fusion

Consider a sequence of RGB images $\{I_n\}_{n:0\dots N}$. Let $\boldsymbol{d}_{n,m}: \Omega \to \mathbb{R}^2$ be a *displacement field* defined on the continuous rectangular domain Ω, such that for every $\boldsymbol{x} \in \Omega$ it corresponds a *displacement vector* $\boldsymbol{d}_{n,m}(\boldsymbol{x}) \in \mathbb{R}^2$ for the ordered pair of images $\{I_n, I_m\}$. Given a *reference image*, say I_0, dense point tracking is compactly represented by $\boldsymbol{d}_{0,m}\ \forall m: 1 \dots N$ (*from-the-reference* correspondences), *i.e.*, the set of displacement fields from I_0 to the subsequent frames I_m. Instead, for propagating (pulling) information present at a key reference frame to the rest of the sequence it is often more natural to deal with $\boldsymbol{d}_{n,0}\ \forall n: 1 \dots N$ (*to-the-reference* correspondences). This is illustrated in Fig. 1.

We address the problem of estimating from-the-reference as well as to-the-reference long-term displacement fields from elementary optical flow fields. Temporal integration of successive optical flow fields is possible but flow estimation errors are inevitably accumulated through this process. A solution would be to estimate the direct displacements between the reference frame and the other frames. However the longer the distance in time between two frames, the more ambiguous the matching process. So-called large displacement dense matching algorithms deal either with fast motion between consecutive frames [5] (but are not at all oriented to finding point correspondences along hundreds of frames) or assume parametric models [27] also constrained to limited frame distances. However, matching non-consecutive (time distant) frames can still be very useful as its accuracy much depends on inter-frame motion range: indeed one observes that for short/mid-term dense point matching, some regions of the image are better matched by concatenating consecutive motion vectors, while for others a direct matching is preferred (e.g., if displacement between consecutive frames is small). Then, the idea is to consider multiple displacement fields with various inter-frame distances in order to have the best vectors available among all the candidates. The process is carried out in three phases: first, *elementary* optical flow fields with various inter-frame distances (called *steps*) are estimated between arbitrary frames $\{I_i, I_j\}_{i,j:0\dots N}$. Then, considering a pair $\{I_n, I_m\}$, various candidate *displacement fields* $\boldsymbol{d}_{n,m}$ are computed by concatenating different elementary fields. Finally the optimal displacement field $\boldsymbol{d}^*_{n,m}$ is obtained by merging these candidate fields. This is called *Multi-Step Fusion (MSF)*.

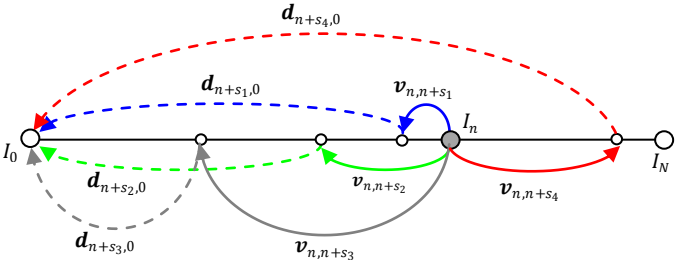


Figure 2. Multi-step point correspondence. The displacement from frame I_n to frame I_0 can be generated following different "paths" according to the available elementary motion fields (solid lines) and the previously estimated long-term displacements (dashed lines).

Define an initial set of possible step values $S = \{s_1, \dots, s_Q\}$ with $s_k \in [-N, -1] \cup [1, N]$. Now, considering the pair $\{I_n, I_0\}$, (respectively, the current and reference frames), let $S_n \subset S$ be the plausible subset of steps $S_n = \{s_k \in S / -n \leq s_k \leq N - n\}$ with $|S_n| = Q_n$. For the given n we consequently consider an input set $\boldsymbol{M}_n$ of Q_n elementary optical flow fields $\boldsymbol{v}_{n,t}$ (between frames n and t): $\boldsymbol{M}_n = \{\boldsymbol{v}_{n,n+s_k}\}_{s_k \in S_n}$. For each input optical flow within this set, one can compute a displacement field between I_n and I_0 resulting from the combination of the elementary field $\boldsymbol{v}_{n,n+s_k}$ and the displacement $\boldsymbol{d}_{n+s_k,0}$ available between I_{n+s_k} and I_0. For each $\boldsymbol{x}$ we thus write:

$$\boldsymbol{d}^k_{n,0}(\boldsymbol{x}) = \boldsymbol{v}_{n,n+s_k}(\boldsymbol{x}) + \boldsymbol{d}_{n+s_k,0}\left(\boldsymbol{x} + \boldsymbol{v}_{n,n+s_k}(\boldsymbol{x})\right). \quad (1)$$

In this manner we generate different candidate displacements or *paths* (Fig. 2) among which we aim at deciding the optimal for each pixel $\boldsymbol{x}$. The process runs a first pass sequentially from frames I_1 to I_N relying on displacement fields estimated at previous frames. In this case, considered step values are negative (as s_1, s_2 and s_3 in Fig. 2). We propose to extend the set of available steps proposed in [9] to positive steps (e.g., $s_4 > 0$ in Fig. 2), by considering a second pass from frames I_{N-1} to I_1that takes into account new candidates corresponding to frames m (m>n in our example) whose displacement field $\boldsymbol{d}_{m,0}$ was not yet available during the first pass. The novelty is that a correspondence can be built by combining both forward and backward intermediate displacements. Not just a matter of adding more candidates, the ability of moving back and forth permits to handle more appropriately temporary occlusions and motion discontinuities. The selection of the optimal path for all the points of the grid for a pair $\{I_n, I_0\}$ is achieved via a global optimization stage that fuses all the candidate fields into a single optimal displacement field $\boldsymbol{d}^*_{n,0}$. While a purely discrete model, such as a Potts-like energy on the path labels, may seem suitable as proposed in [9, 27] such a label-based regularization does not necessarily translate in spatial smoothness of motion. Instead, we propose to minimize:

$$E_{n,0}(\boldsymbol{K}) = \sum_{\boldsymbol{x}} C_{n,0}\left(\boldsymbol{x}, \boldsymbol{d}^{k_{\boldsymbol{x}}}_{n,0}(\boldsymbol{x})\right) + \sum_{<\boldsymbol{x},\boldsymbol{y}>} \alpha_{\boldsymbol{x},\boldsymbol{y}} \cdot \left\| \boldsymbol{d}^{k_{\boldsymbol{x}}}_{n,0}(\boldsymbol{x}) - \boldsymbol{d}^{k_{\boldsymbol{y}}}_{n,0}(\boldsymbol{y}) \right\|_1, \quad (2)$$

w.r.t. $\boldsymbol{K} = \{k_{\boldsymbol{x}}\}$, a complete labeling of the image grid, where each label indicates one of the candidate paths; $C_{n,0}(\boldsymbol{x}, \boldsymbol{d})$ is a matching cost between location $\boldsymbol{x}$ in image I_n and location $\boldsymbol{x} + \boldsymbol{d}$ in I_0. Meanwhile, regularization is enforced between the displacement vector values rather than the label values: $< \boldsymbol{x}, \boldsymbol{y} >$ is a pair of neighbouring image

locations according to 8-point connectivity. $\alpha_{x,y}$ accounts for colour and elementary motion (step 1) spatial similarities (see section 4 for more details). Standard graph-cut optimization techniques cannot be applied since the resulting energy does not meet certain necessary conditions [13]. We then apply the method recently presented in [15, 16] in the context of instantaneous optical flow estimation by flow fusion. It results that for each point we obtain the best path label k_x, and this in turn gives the optimal long-term correspondence vector $\boldsymbol{d}^*_{n,0}(\boldsymbol{x}) = \boldsymbol{d}^{k_x}_{n,0}(\boldsymbol{x})$. Of course, this can be generalized to any reference frame, and the application to the *from-the-reference* displacement fields is straightforward.

3 Multilateral spatio-temporal filtering

Once forward and backward displacement/trajectory fields exit the multi-step fusion stage, they can be advantageously combined in a mutual refinement step. Actually, forward and backward fields $\boldsymbol{d}_{0,n}$ and $\boldsymbol{d}_{n,0}$ that have been estimated independently carry complementary, or sometimes contradictory, information. In addition, the trajectory features provided by the forward fields are also taken into account: the set of forward vectors $\boldsymbol{d}_{0,n}(\boldsymbol{x})$ describes the trajectory of point $\boldsymbol{x}$ in frame I_0 along the sequence. The iterative filtering described below is preceded by occlusion detection (adapted from the *occlusion constraint* (OCC) method described in [10]) and inconsistency evaluation (left/right disparity checking in [10], here applied to forward/backward motion fields).

For all pairs $\{I_0, I_n\}$, forward displacement fields $\boldsymbol{d}_{0,n}$ are first spatio-temporally filtered considering the trajectories of spatial neighbouring pixels in the reference frame I_0. This step increases the consistency in terms of trajectory behaviour for neighbouring pixels. Then, forward and backward displacement fields $\boldsymbol{d}_{0,n}$ and $\boldsymbol{d}_{n,0}$ are jointly processed via multilateral filtering which propagates iteratively the refinement from the forward (respectively backward) direction to the backward (respectively forward) direction.

The "trajectory" aspect of the forward fields is considered in two ways. First, a trajectory similarity weight between neighbouring pixels in I_0 is introduced. Second, displacement fields in neighbouring frames are taken into account in the filtering process. Forward displacement fields $\boldsymbol{d}_{0,n}(\boldsymbol{x})$ are iteratively filtered considering the neighbouring forward vectors $\boldsymbol{d}_{0,m}(\boldsymbol{y})\big|_{m:n-\Delta \dots n+\Delta}$ where Δ defines a temporal window. This first step is defined as follows:

$$\widetilde{\boldsymbol{d}}^{\boldsymbol{FW}}_{0,n}(\boldsymbol{x}) = \frac{\sum_{m=n-\Delta}^{m=n+\Delta} \sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{x}\}}} w^{\boldsymbol{xy}}_{traj} \cdot w^{\boldsymbol{xy}}_{0,m} \cdot (n/m) \cdot \boldsymbol{d}_{0,m}(\boldsymbol{y})}{\sum_{m=n-\Delta}^{m=n+\Delta} \sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{x}\}}} w^{\boldsymbol{xy}}_{traj} \cdot w^{\boldsymbol{xy}}_{0,m}}, \tag{3}$$

where $\mathcal{F}_{\{\boldsymbol{x}\}}$ defines a spatial neighbourhood of $\boldsymbol{x}$. Each vector in neighbouring frames $(m \neq n)$ is weighted by a scaling factor n/m in order to make the input motion fields $\boldsymbol{d}_{0,m}(\boldsymbol{y})\big|_{m:n-\Delta \dots n+\Delta}$ that correspond to different temporal distances, comparable. $w^{\boldsymbol{xy}}_{s,t}$ is a weight that links points $\boldsymbol{x}, \boldsymbol{y}$ at frame I_s based on their motion to frame I_t:

$$w^{\boldsymbol{xy}}_{s,t} = \rho_{s,t} \cdot e^{-\left(\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{\gamma} + \frac{\sum_{c\in\{r,g,b\}} |I^c_S(\boldsymbol{x}) - I^c_S(\boldsymbol{y})|}{\varphi} + \frac{\sum_{c\in\{r,g,b\}} \left| I^c_S(\boldsymbol{y}) - I^c_t\left(\boldsymbol{y}+\boldsymbol{d}_{s,t}(\boldsymbol{y})\right) \right|}{\theta} \right)}. \tag{4}$$

This weight combines spatial distance, colour similarity and matching cost. It involves $I_s^c(\boldsymbol{x})$ which corresponds to a RGB component at location $\boldsymbol{x}$ in image I_s. $\rho_{s,t}$ is a binary value that is 1 only if the point was not detected as occluded. γ, φ and θ are positive constants used to adjust the weight components. For uniform areas we set $\gamma \to \infty$ and φ is increased to limit the effect of pixels with a different colour value. Pixels in I_s that belong to uniform areas are those for which:

$$\sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{x}\}}} \exp\left[-\frac{\left(\sum_{c\in\{r,g,b\}}|I_S^c(\boldsymbol{x})-I_S^c(\boldsymbol{y})|\right)^2}{\xi}\right] > 0.5 \qquad (\xi > 0). \tag{5}$$

The weight $w_{traj}^{\boldsymbol{xy}}$ derives from the similarity of the trajectories that support the two currently compared forward vectors. It is defined as:

$$w_{traj}^{\boldsymbol{xy}} = \exp\left[-\frac{\sum_{m=1}^{m=N}\|\boldsymbol{d}_{0,m}(\boldsymbol{x})-\boldsymbol{d}_{0,m}(\boldsymbol{y})\|_2^2}{\psi}\right] \qquad (\psi > 0). \tag{6}$$

Once forward displacement fields have been spatio-temporally filtered, joint multilateral filtering (both forward/backward and backward/forward) is performed. Noting $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{d}_{0,n}(\boldsymbol{x})$, the updated forward displacement field is:

$$\widetilde{\boldsymbol{d}}_{0,n}(\boldsymbol{x}) = \frac{\sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{x}\}}} w_{0,n}^{\boldsymbol{xy}}\cdot\widetilde{\boldsymbol{d}}_{0,n}^{\boldsymbol{FW}}(\boldsymbol{y}) - \sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{z}\}}} w_{n,0}^{\boldsymbol{zy}}\cdot\boldsymbol{d}_{n,0}(\boldsymbol{y})}{\sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{x}\}}} w_{0,n}^{\boldsymbol{xy}} + \sum_{\boldsymbol{y}\in\mathcal{F}_{\{\boldsymbol{z}\}}} w_{n,0}^{\boldsymbol{zy}}}. \tag{7}$$

The weights are defined as in (4) except that a motion vector similarity term replaces the trajectory similarity (not available for the backward direction). The backward vector is filtered analogously with $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{d}_{n,0}(\boldsymbol{x})$.

The three steps of the whole spatio-temporal filtering method (spatio-temporal filtering and forward/backward and backward/forward joint multilateral filtering) are iterated a given number of times. Every 3 iterations, the whole process is applied to the totality of the vectors while, for the other iterations, it is limited to those for which the inconsistency value is above a threshold. In this way, we give more confidence to consistent values in a soft manner.

An *a posteriori* choice between unfiltered and filtered vectors is performed with respect to the matching cost while encouraging filtering to some extent. We apply the global optimization proposed in (2) in order to fuse unfiltered and filtered vectors.

4 Experiments

We propose different types of experiments to assess the performance of our method along with comparisons with state-of-the-art approaches. We focus on two video sequences that combine a rich set of interesting characteristics (Fig. 3): *AmeliaRetro* (courtesy of *Dolby*®) ($1920 \times 1080 \times 100$ frames featuring zooming, occlusions, spatial lighting variation); and *Newspaper* ($1024 \times 768 \times 100$ frames featuring temporary occlusions, fixed background, appearing background object, shadows, low colour contrast between different motion regions).

Parameter specification. We define $C_{n,0}(\boldsymbol{x}, \boldsymbol{d})$ in (2) as the mean absolute difference (MAD) of pixel colour values between image windows of size 5×5. For 8-bit colour components, the value of MAD is then truncated to a maximum of 128 in order to

Figure 3. Video sequences used in the experiments. Top: frames 0,25,75,100 of *AmeliaRetro,* Bottom: frames 0,25,50,75,100 of *Newspaper* (cropped).

robustify the measurement. Moreover, the effect of illumination variations and shadows is attenuated by normalizing each colour pixel of the input images by a local mean intensity. Meanwhile, the parameter $\alpha_{x,y}$ in (2) is defined as $\alpha_{x,y} \equiv \alpha^n_{xy} = 20 \cdot \alpha^n_{xy}\big|_{color} \cdot \alpha^n_{xy}\big|_{motion}$. First, $\alpha^n_{xy}\big|_{color} = e^{-\|c^n_x - c^n_y\|_1/\sigma}$ with $\boldsymbol{c}^n_{\boldsymbol{x}}, \boldsymbol{c}^n_{\boldsymbol{y}}$ the 3-channel colour vectors at locations $\boldsymbol{x}$ and $\boldsymbol{y}$, for image n, respectively, and $\sigma = 300$. This enforces smoothness of the motion vectors assigned to nearby pixels with similar colour. Second, $\alpha^n_{xy}\big|_{motion} = e^{-\|\boldsymbol{v}_{n,n\pm1}(\boldsymbol{x}) - \boldsymbol{v}_{n,n\pm1}(\boldsymbol{y})\|_1/10}$ with $\boldsymbol{v}_{n,n\pm1}(\boldsymbol{x})$ the motion vector from the (forward or backward) input optical flow between consecutive images (with step 1). Instantaneous optical flow is a reliable measure that gives useful information about motion discontinuities. Though it is true that long-term displacements are also valuable (and complementary), we cannot rely on them at this early stage.

Regarding multilateral filtering, the spatial and temporal windows are respectively of size 7×7 and 3. The number of iterations has been empirically set to 19. Moreover, $\gamma = 200$, $\varphi = 600$ (1000 if the corresponding pixel belongs to a uniform area), $\theta = 600$, $\xi = 200$, $\psi = 5 \times N$. The threshold for the inconsistency evaluation equals to 1 pixel. Finally, the global optimization described in (2) is applied to fuse unfiltered and filtered vectors with $\alpha_{x,y} \equiv 20 \cdot \alpha^n_{xy}\big|_{color}$.

Input optical flows. The set of input elementary optical flow fields is manually selected as to handle a rich variety of situations within each video sequence. Though it may depend on the video content, a basic set of candidate steps is $S = \{\pm1, \pm2, \pm5, \pm10, \pm20, \pm30, \pm40, \pm50, \pm100\}$. However adaptation of this set to each shot might be useful. In our experiments, these motion fields are estimated by means of an adapted 2D version of the 1D disparity estimator described in [21], but any estimator should do. For *AmeliaRetro,* due to the predominance of camera motion, we have also complemented the set of non-parametric optical flow fields with additional affine motion fields for steps $\{\pm10, \pm20, \pm30, \pm50, \pm80\}$. The algorithm is indeed clearly able to handle several candidates for a same given step, possibly estimated with different methods or parameters.

Computation Time. In order to assess the computation time of the process chain we have conducted an experiment given an input sequence with 100 frames of 400×400 pixels. On average, it takes ~ 2 seconds per frame and per candidate path to perform the construction of the energy and the global optimization. That is, with c candidate paths, the

fusion process takes $\sim 2c$ seconds. Regarding multilateral filtering applied with the parameters described above, around 90 seconds per frame are required.

Comparisons. We compare our multi-step fusion method w.r.t. other state-of-the-art approaches: the *TV-L1* optical flow method [28], *Large Displacement Optical Flow (LDOF)* [5, 25], basic *Multi-Step* via *Graph-cuts (MS-GC)* [9] and *ParticleVideo (PV)* [22]. The first two provide dense point tracks by motion integration; *MS-GC* is our baseline method for the multi-step approach; *PV* estimates a sparse set of tracks. We also test different versions of our approach: single step (step = 1) optical flow estimation from [21] (*STEP1),* multi-step fusion without filtering *(MSF)* and multi-step fusion with spatio-temporal filtering (*MSF+STF).*

Trajectory evaluation. We have picked 8 points in I_0 of *AmeliaRetro*, carefully selected as to account for textured and non-textured areas. We manually generated the ground truth trajectories along the 100 frames. We then measured the frame-by-frame position error for several methods as depicted in Fig. 4 (top). For each method, we plot the median error among the 8 points at each instant. We draw the following remarks from the plot: a) the three methods based on optical-flow integration (*TV-L1*, *LDOF*, *STEP1*) are the worst performing, supporting our claim that high precision in instantaneous motion estimation does not guarantee long-term tracking accuracy; b) Multi-step methods start to perform better after $\sim$30 frames, duration that is coherent with the maximum track length used in [4, 17, 25, 29] c) the most accurate method is *MSF+STF*, specially noting how the position error at frame 63 is as small as in frame 7. The optimal combination of short and long term matching did its job reducing the drift.

For *Newspaper*, we analyze the complex situation of a temporary occlusion, as observed in Fig. 4 (bottom), where the arm and the cup occlude the chest. A total of 19 points were manually tracked, equally spaced by 10 pixels on a vertical line that passes through the chest and the hand. While for single step methods it is impossible to estimate the trajectories of the occluded pixels after the occlusion (finally attaching all the tracks to the motion of the hand, which obliges to stop the trajectory), the multi-step fusion algorithm is able to circumvent the problem thanks to the long-step input displacement fields. Moreover, track segments before and after the occlusion are naturally linked together as each position refers to the same reference point. The filtering step improves the temporal consistency of trajectories.

An ingenious way of quantitatively assessing the quality of the estimated trajectories is by mirroring a sequence in time, *i.e.*, constructing $\{I_n\}_{n:0\ldots N\ldots 0}$ and checking the symmetry of the track [22]. For a given point, the departing location is known to be identical to the arriving position. We go further by testing the same condition for all the pairs of mirrored instants. The *AmeliaRetro* sequence was cropped to a meaningful area of 768×675 pixels and mirrored taking the first 50 frames to generate a new 100 frame video: *AmeliaRetroMirror*. Our method is tested on this sequence and compared to *PV* and *MS-GC*. In Fig. 5, we observe the improvement obtained in terms of precision. Moreover, multi-step methods obtain the full-length tracks for 100% (518400) of the image points in I_0 while *PV* is only able to estimate 0.2% full-length tracks, and initially selects only 0.5% (2610) points in the first image. Higher accuracy together with full density clearly shows the benefits of our approach.

Long-term image warping. The second experiment consists in reconstructing the reference image I_0 from each image I_n of a video sequence exploiting point correspondences. This is a very challenging task which permits to obtain a global measure

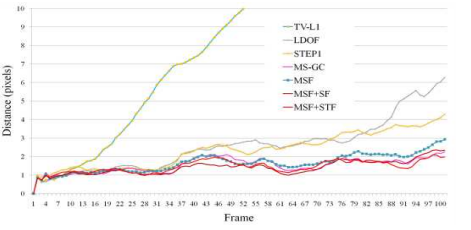
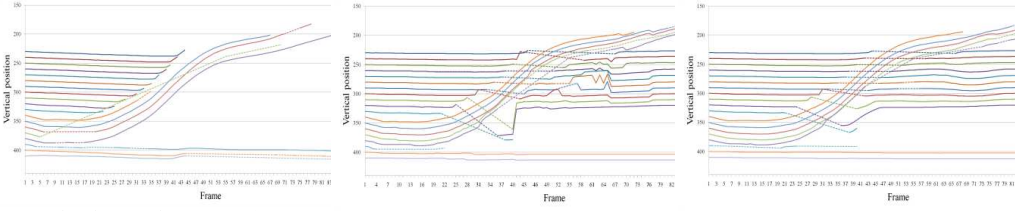


Figure 4. *Top.* Median position error for the 8 ground-truth trajectories for *AmeliaRetro*. *Bottom.* Vertical component of the 10 estimated tracks (see text) for *Newspaper* with different methods. The points were selected equally spaced on a vertical line that passes through the chest and the arm.



Motion integration *(TV-L1)* [28] *MSF* *MSF+STF*

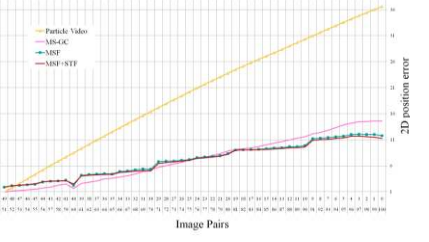


Figure 5. Position error for *AmeliaRetroMirror*. Each data point corresponds to the pair of time instants $[\{49,51\};\{48,52\};\ldots;\{0,100\}]$. Our method shows a better accuracy tracking 100% of the points compared to *Particle Video* [22], which is only able to completely track 0.2% of the points.

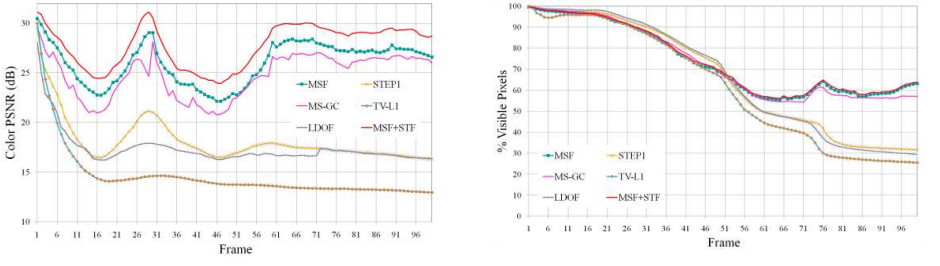


Figure 6. Image warping error and occlusion handling using forward correspondences. Left: quality of reconstruction of the reference image I_0 from each subsequent frame $I_{n>0}$ for *AmeliaRetro*. Right: percentage of visible points detected by each method for *Newspaper*.

of the performance of an algorithm. Indeed, large colour differences clearly show defective correspondences. This is achieved for *AmeliaRetro* by copying the colour values from I_n according to the displacement field $\boldsymbol{d}_{0,n}(\boldsymbol{x})$. Note that this corresponds to a *from-the-reference* strategy. Then we compute the colour PSNR in a block within the dress of the lady of each reconstructed reference image w.r.t. the original reference image I_0. In Fig. 6 (left) we observe that multi-step approaches are clearly better for reconstruction than standard optical flow integration. And among them, the improvement of *MSF* methods is significant especially w.r.t. the baseline multi-step *MS-GC* method. The PSNR assessment for the to-the-reference strategy (reconstruction of I_n from I_0) gives also good results. The same experiment was conducted for the sequence *Newspaper*. However given that the colour of moving regions is basically the same (blue) the curves were not meaningful for assessing the accuracy of the correspondence estimation. On the other hand, it is

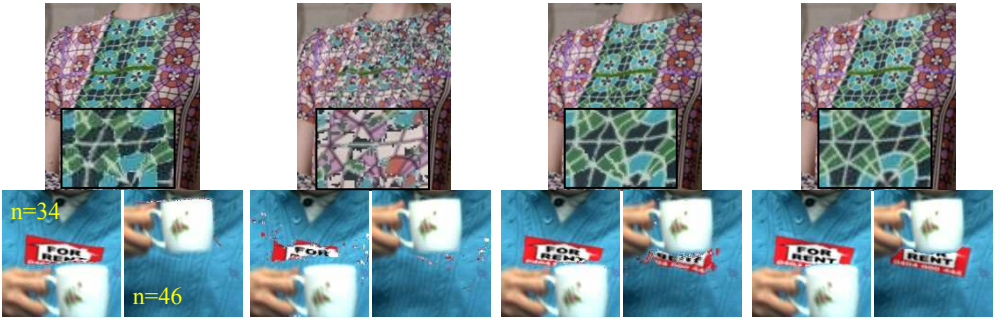


Figure 7. Video editing examples. For *AmeliaRetro* (top) the reconstruction of the texture is improved with our method. For *Newspaper* (bottom) we highlight the consistency of the insertion before and after the occlusion only for *MSF+STF*.

interesting to show the behaviour of each method in front of the temporary occlusions caused by the arm and cup. We thus plot in Fig. 6 (right) the percentage of visible points detected by each method along the sequence. This illustrates the fact that our method is able to recover reappearing points while for single step methods, the number of visible points decreases monotonically.

5 Applications

Video editing. Once we have a set of dense long-term correspondences that link every point of the sequence to the reference frame, the applications in the context of video editing are unlimited. A typical problem is the insertion of external graphical elements on real surfaces within the video. With this regard, we present two results for *AmeliaRetro* and *Newspaper* (Fig. 7). The first consists in changing the colour of a part of the dress of the lady at frame 100 and then propagating this change by using the to-the-reference (forward in this case) fields to the remaining 100 frames. We compare the methods *STEP1* (which was the best result among single step methods), *MS-GC, MSF+STF,* and, as well, Coherency Sensitive Hashing [14] using their estimated patch correspondences. Secondly, we have inserted a logo at frame I_0 of *Newspaper*, which is then automatically inserted in the other frames. Note how the large occlusion by the arm can be overcome only by multi-step methods. However the accuracy of *MSF+STF* is clearly better than *MS-GC*. Note the consistency before and after the occlusion. Moreover, we have taken advantage of the reliable point correspondences in order to compute a brightness gain for each point between the reference and each frame. This permits to insert the element more realistically over a shadowed area.

Key-frame based video segmentation. Let's assume that the user provides a dense segmentation map for a given reference frame. For each grid location $\boldsymbol{x}$ of each non-reference frame of the sequence, and if it is not detected as occluded, we determine its corresponding position in the reference frame. If this position is within the image boundaries, the label of the nearest pixel is given to $\boldsymbol{x}$. At this stage, occluded pixels remain unlabelled. Note that this label propagation process can be easily adapted to use more than one single reference segmentation map. If a conflict appears between the labels propagated at the same pixel $\boldsymbol{x}$ from different reference frames, we simply solve it by

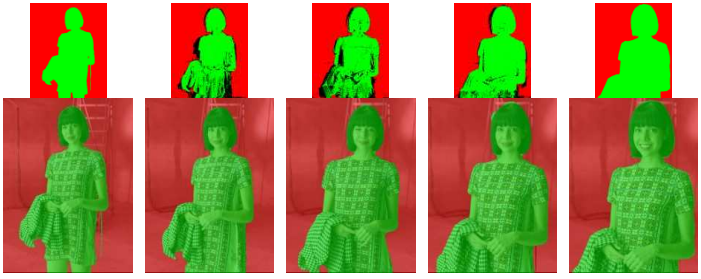


Figure 8. From label propagation to dense segmentation. Frames 0, 25, 50, 75 and 100 are shown. The user provided reference segmentation maps for frames 0 and 100.

assigning $\boldsymbol{x}$ to the label corresponding to the lowest colour matching cost. Dense segmentation (Fig. 8) may then be obtained using standard segmentation tools. Precisely, to refine the maps obtained by label propagation and to assign a label at the occluded pixels, we perform a graph-cut minimization of a cost function which is the sum of two standard terms. The first term is a colour data penalty term of assigning label l at pixel $\boldsymbol{x}$. It is set as the negative log-likelihood of colour distribution of the video region l. This distribution consists of the Gaussian mixture model in the RGB space computed on the regions l in the reference segmentation maps. The second term is the standard contrast sensitive regularization term defined in [3]. Note the slight rotation of the girl that results in self-occlusions w.r.t. both key-frames. Temporal consistency of the segmentation could be reinforced by adding an explicit constraint based on available motion fields estimated between consecutive frames.

6 Conclusion

We presented a new algorithm for estimating dense correspondence fields between arbitrary distant frames in long video shots. It is based on the combination, then optimal merge, of several intermediate candidate displacement fields, including forward, as well as backward, elementary optical flows. The notion of trajectory is then explicitly taken into account in a novel spatio-temporal filtering step. Compared to state-of-the art approaches, the fields resulting from our approach present an improved accuracy, particularly for large motions and in presence of temporary occlusions.

A point that would deserve further investigation is the automatic selection of both the reference frames and the input set of candidate steps depending on the considered shot. They have indeed to be set properly as each shot contains its own motion peculiarity. For reference frames, the aim would be for instance to automatically identify the smallest set of frames that contain alone all the regions visible along the shot.

References

[1] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sanchez. Symmetrical dense optical flow estimation with occlusion detection. ECCV, 2002.

[2] C. Barnes, E. Shechtman, A. Finkelstein and D. B. Goldman. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. SIGGRAPH, 2009.

[3] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. ICCV, 2001.

[4] M. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. ECCV, 2010.

[5] T. Brox and J. Malik. Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation. *PAMI*, 33(3): 500-513, 2011.

[6] A. Buchanan and A. Fitzgibbon. Combining local and global motion models for feature point tracking. CVPR, 2007.

[7] X. Cao, Z. Li and Q. Dai. Semi-Automatic 2D-to-3D Conversion Using Disparity Propagation. *IEEE Trans. on Broadcasting*, 57(2): 491-499, 2011.

[8] T. Corpetti, É. Mémin and P. Pérez. Dense Estimation of Fluid Flows. *PAMI*, 24(3): 365-380, 2002.

[9] T. Crivelli, P.-H. Conze, P. Robert and P. Pérez. From optical flow to dense long term correspondences. ICIP, 2012.

[10] G. Egnal and R. Wildes. Detecting binocular half-occlusions: empirical comparisons of five approaches. *PAMI*, 24(8): 1127-1133, 2002.

[11] R. Garg, A. Roussos and L. Agapito. Robust Trajectory-Space TV-L1 Optical Flow for Non-rigid Sequences. EMMCVPR, 2011.

[12] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, vol 17, 185-203, 1981.

[13] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *PAMI*, 26(2): 147-159, 2004.

[14] S. Korman and S. Avidan. Coherency Sensitive Hashing. ICCV, 2011.

[15] V. Lempitsky, S. Roth and C. Rother. FusionFlow: Discrete-continuous optimization for optical flow estimation. CVPR, 2008.

[16] V. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion Moves for Markov Random Field Optimization. *PAMI*, 32(8): 1392-1405, 2010.

[17] J. Lezama, K. Alahari, J. Sivic and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. CVPR, 2011.

[18] C. Liu, J. Yuen and A. Torralba. SIFT Flow: Dense Correspondence across Scenes and Its Applications. *PAMI*, 33(5): 978-994, 2011.

[19] N. Papenberg, A. Bruhn, T. Brox, S. Didas and J. Weickert. Highly Accurate Optic Flow Computation with Theoretically Justified Warping. *IJCV*, 67(2): 141-158, 2006.

[20] A. Rav-Acha, P. Kohli, C. Rother and A. Fitzgibbon. Unwrap Mosaics. SIGGRAPH, 2008.

[21] P. Robert, C. Thébault and P.-H. Conze. Disparity-compensated view synthesis for s3D content correction, In Proc. SPIE Stereoscopic Displays & Applications XXIII, 2012.

[22] P. Sand and S. Teller. Particle Video: Long-Range Motion Estimation Using Point Trajectories. *IJCV*, 80(1): 72-91, 2008.

[23] J. Shi and C. Tomasi. Good Features to Track. CVPR, 1994.

[24] D. Sun, S. Roth and M. J. Black. Secrets of optical flow estimation and their principles. CVPR, 2010.

[25] N. Sundaram, T. Brox and K. Keutzer. Dense point trajectories by GPU-accelerated large displacement optical flow. ECCV, 2010.

[26] H. Wang, A. Klaser, C. Schmid and C. Liu. Action recognition by dense trajectories. CVPR, 2011.

[27] J. Wills, S. Agarwal and S. Belongie. A Feature-based Approach for Dense Segmentation and Estimation of Large Disparity Motion. *IJCV*, 68(2): 125-143, 2006.

[28] C. Zach, T. Pock and H. Bischof. A duality based approach for realtime TV-L1 Optical Flow, *Pattern Recognition*, 4713(22): 214-223, 2007.

[29] G. Zhang, J. Jia, W. Hua and H. Bao. Robust Bilayer Segmentation and Motion/Depth Estimation with a Handheld Camera. *PAMI*, 33(3): 603-617, 2011.