Determining occlusions from space and time image reconstructions

Juan-Manuel Pérez-Rúa[1,2], Tomas Crivelli[1], Patrick Bouthemy[2], and Patrick Pérez[1]

[1]Technicolor, Cesson Sévigné, France [2]Inria, Centre Rennes - Bretagne Atlantique, France

Abstract

The problem of localizing occlusions between consecutive frames of a video is important but rarely tackled on its own. In most works, it is tightly interleaved with the computation of accurate optical flows, which leads to a delicate chicken-and-egg problem. With this in mind, we propose a novel approach to occlusion detection where visibility or not of a point in next frame is formulated in terms of visual reconstruction. The key issue is now to determine how well a pixel in the first image can be "reconstructed" from co-located colors in the next image. We first exploit this reasoning at the pixel level with a new detection criterion. Contrary to the ubiquitous displaced-frame-difference and forward-backward flow vector matching, the proposed alternative does not critically depend on a precomputed, dense displacement field, while being shown to be more effective. We then leverage this local modeling within an energy-minimization framework that delivers occlusion maps. An easy-to-obtain collection of parametric motion models is exploited within the energy to provide the required level of motion information. Our approach outperforms state-of-the-art detection methods on the challenging MPI Sintel dataset.

1. Introduction

Detecting occluded areas at each instant of a video sequence is of utmost interest for many computer vision applications. In fact, even though occlusion detection is mostly associated with the problem of computing inter-image correspondences (optical flow for monocular vision, or disparity map in stereo vision), it is very informative on its own. Among other applications, occlusion-based reasoning has been applied to contour and object tracking [21, 44], segmentation of multiple objects [39], action recognition [37], pose estimation [36], and depth ordering [26].

In spite of the usual association between motion field estimation and occlusion detection, it is worth noting that physical motion within the scene by itself does not determine if an element is hidden at a given instant. An additional factor is needed in the equation, that is, the observer point of view, or in other words, the observed 2D visual representation of the real 3D world, *i.e.*, the image. This is a well-known fact and limitation of the optical flow as a representation of physical motion [35]. When working with a succession of discrete-in-time and discrete-in-space 2D images, one can define that a point of a given image is occluded in the next (or other) image of the sequence, if it is not visible by the observer in the latter.

Many state-of-the-art approaches tackle occlusion detection based on the following question: *Does an image point have a correspondent in the other image that can be confidently identified as physically identical?* In practice, this is indeed evidenced, either implicitly or explicitly, by a wide range of formulations that consider the problem of occlusion detection as inseparable of displacement estimation [2, 13, 18, 32, 41]. This simple question leads however to obstacles in formulating the problem. First, true dense correspondences between images are not easily obtained, especially on occluded pixels where the optical flow (or disparity, likewise) is not well defined. Even in non-occluded areas, rapid changes in appearance and scale make the definition and the estimation of unique correspondences difficult.

We strive for an alternative, more direct, approach to occlusion detection. We adopt an image reconstruction viewpoint that frees us, to a large extent, from the need of jointly estimating an accurate, dense motion field. Instead, we only make use of "plausible" motions simply extracted from the image pair.

The main idea is to assess whether or not pixel appearance can be equally well explained by its spatial neighborhood in the same frame and by suitable co-located pixels in the other frame. If not, this point is likely to be occluded in the next image. In this way, occlusion detection can be sought as independent of the knowledge of an accurate, univocal motion field between the two images. It suffices to exploit loosely the spatiotemporal coherency in order to select a suitable spatiotemporal neighborhood.

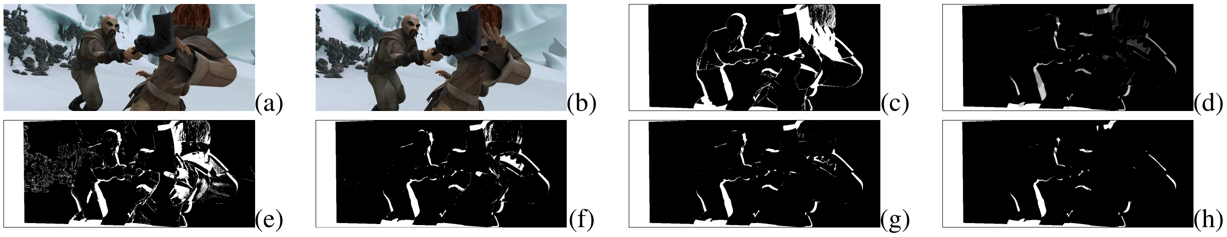

Figure 1. **Occlusions from color inconsistencies along *true* flow**. (a-b) Two successive frames of the clean *ambush_2* sequence from the *MPI Sintel* dataset [10]. (c) Occlusion ground truth with occlusions shown in white; (d) Norm of the color difference between points matched by the true flow (2D projection of the known 3D motions); (e)-(h) Occlusion maps obtained by thresholding at 0.015, 0.050, 0.100 and 0.250 respectively, the *min-max* normalized (between 0 and 1) color differences along the flow. They are all unsatisfactory (low precision and/or low recall).

The paper is organized as follows. In Section 2 we discuss relevant literature and motivate the need for a different approach to occlusion detection. We then devote Section 3 to introducing the ideas behind our novel approach, whose formulation is given in Section 4. We report experimental results, including extensive comparisons to recent occlusion detection methods, in Section 5. We provide concluding remarks in Section 6.

2. On true motion and occlusion models

Seeing occlusion detection as only part of a joint motion-occlusion problem requires (1) to model accurately visual motion and (2) to model occlusions in the light of this motion. As we will show, even if the true motion field is known, approaches of this type might fail to produce accurate occlusion maps. As a consequence, we argue that motion should remain an auxiliary variable and not the main object of interest. This idea deeply contrasts with the reasonings one can find in the literature.

A popular idea is that an occlusion is a violation of the optical flow constraint: *“Occlusions generally become apparent when integrated over time because violations of the brightness-constancy constraint of optical flow accumulate in occluded areas*” [12].

Other authors make similar claims, without referring to optical flow integration but stating instead that, under Lambertian reflections and the constant illumination assumption, a brightness change between corresponding points indicates an occlusion of the point in the second image [4, 40].

Another assertion is that flow errors occur in occlusion areas or that an occlusion is an explanation of motion mismatching: *“. . . the most probable reasons for such a situation* [flow mismatching] *is an occlusion problem or an error in the estimated matching”* [2]. This notion has also been exploited by other works [13, 22, 43] where forward-backward flow inconsistency is used to detect occlusions.

Similarly to the brightness conservation constraint, some authors have proposed that a point is occluded if it switches from one motion layer or segment to another between consecutive frames: *“To consistently reason about occlusions, we examine the layer assignments* [of two points in consecutive frames] *at locations corresponded by the underlying flow fields”* [30]. In our opinion, this argument falls short for non-planar motions and self-occlusions even when optical flow is allowed to deviate from the assumed parametric motion as in [30]. Ambiguities on occlusion estimation from layer assignment are alleviated by enforcing temporal layer consistency [31].

Relying on a joint motion-occlusion estimation leads to a chicken-and-egg situation, which is handled in alternation: “[The algorithm] *iterates between estimating values for* [occlusion map]*, and optimizing the current optical flow estimates by differential techniques”* [28].

Other approaches include the uniqueness criterion [9], which is known for producing a large amount of false positives [42], and combinations of the criteria explained above. For instance, [15] considers flow symmetry in combination with the violations of the optical flow brightness constancy constraint. Another example is [32] which enforces disparity consistency by labeling points that cannot be matched with another point in the second image as occluded, while proposing a sequential approach that computes occlusion and disparity iteratively. Different views on the related problem of finding occlusion borders, but without determining exactly the occlusion regions are found in [3, 16, 23, 25, 27, 33].

Doubtless, the underlying motion is indeed helpful to find occlusions. This is further confirmed by the work of Humayun *et al.* [14] where a large amount of features were used to train a random forest, giving a high importance to motion-based features. In the same way, knowing the occlusion labeling of image pixels clearly helps motion estimation. Specifically, it better guides regularization and smoothing. Many optical flow methods try to deal with occlusions by embedding a discrete state into a continuous numerical scheme [5], or an aggregation framework [13], ending with complex, usually joint formulations that improve motion estimation. Similarly, stereo vision approaches that are formulated as discrete label-selection problems can be naturally extended to handle occlusions by adding a label for the occlusion state [17, 18], relying on efficient discrete energy optimization techniques.

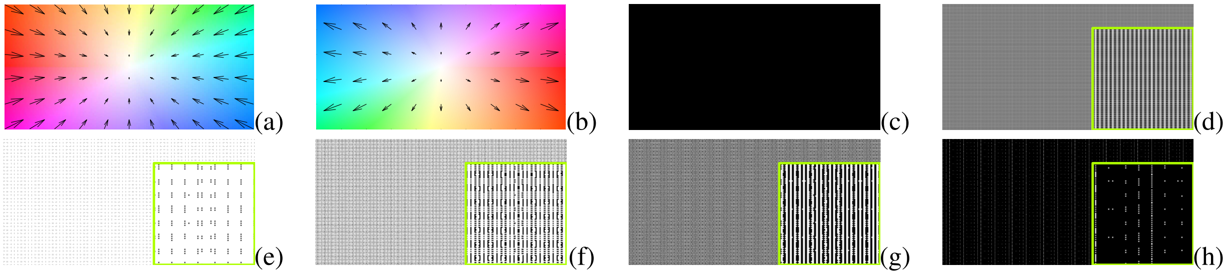

Figure 2. **Fictitious occlusions from forward-backward inconsistencies of divergent flows**. (a)-(b) Forward-backward divergent flows associated to a zoom-in (classic hue-saturation color coding at the pixel level, and subset of motion vectors superimposed for better visualization). (c) True occlusion map, devoid of occlusions; (d) Norm of the forward-backward flow difference between points matched by the true flow. Because of interpolations required to evaluate backward flows at non-pixel positions, differences are space dependent (e)-(h) Occlusion maps obtained by thresholding at 0.100, 0.333, 0.500 and 0.900, respectively, the normalized forward-backward differences along the flow. For better visualization of the error pattern, we zoom-in the resulting images (bottom-left corner). False positives occur all over the image grid.

In order to pin down our claims, let us assume that we know the true motion field for an image pair[1] and let us analyze the most common underlying reasonings for occlusion detection. Figure 1 shows results for the occlusion detection by finding violations of the color constancy assumption along this motion, that is by thresholding the so-called displaced frame difference (DFD). If $\mathbf{x}$ and $\mathbf{x}'$ are locations on image grid Ω of two truly corresponding pixels in color images I_1 and I_2 respectively, $\mathbf{x}$ is declared occluded if

$$\|I_1(\mathbf{x}) - I_2(\mathbf{x}')\| > \varepsilon_c, \tag{1}$$

where ε_c is a threshold, which is proposed with variants in [4, 40]. Even though the image sequence in this example does not contain significant illumination changes nor extra post-processing effects like mist or motion blur, the *color constancy criterion is not robust enough to detect occlusions even if the true motion is known*. This is easily verified visually by looking at regions where the norm of the color difference does not have large enough values even across occluded areas (Fig.1).

A similar experiment can demonstrate that surprisingly, even when the true optical flow is available, the forward-backward flow consistency criteria might fail to accurately capture the real occlusion map (Fig.2). This criterion assumes that given corresponding points $\mathbf{x}$ and $\mathbf{x}'$ and their associated forward and backward flows, $\mathbf{w}_f(\mathbf{x})$ and $\mathbf{w}_b(\mathbf{x}')$, $\mathbf{x}$ is declared occluded if $\|\mathbf{w}_f(\mathbf{x}) + \mathbf{w}_b(\mathbf{x}')\| > \varepsilon_f$, where ε_f is a threshold [2, 7, 13]. Implementation-wise, the backward flow at $\mathbf{x}'$ is obtained by bilinear interpolation, since $\mathbf{x}'$ is generally not an integer grid position. Such a detail by itself generates erroneous flow mismatching which may be enlarged in different situations such as motion discontinuities or zooming. Even if ground-truth motion is available at image grid points, $\mathbf{w}_b(\mathbf{x}')$ may introduce a position drift while going backwards to the first image. Figure 2 shows occlusion maps obtained by this criterion for a synthetic zoom. The errors in the occlusion map can be explained by the grid discretization of the flows, making the forward-backward flow difference grow in several zones of the non-occluded area (whiter pixels in Fig.2), and leading to a large number of false positives.

As the aforementioned two criteria are the most commonly used [2, 4, 7, 13, 40], the amount of errors they can lead to should not be neglected. Reducing the dependency of occlusion detection on flow quality, as we propose, is one answer to this problem.

3. From image reconstructions to occlusion

We consider that the property of being occluded is intrinsic to each one of the points of an image. This means that detecting occlusions may be posed as an independent problem, and not necessarily strongly attached to a per-pixel estimation of motion. We start from the standard concept of an occluded point, represented in the image space through the simplified concept of a pixel, as one that is visible in a first image and not visible in a second image.

A visible to non-visible transition implies a loss of information between the two images. This means that there is a pixel in the first image that cannot be explained using the second image. At a larger scale, to pin things down, suppose a well-defined object present at one instant. The question we ask is: *Can the visual information carried by the object be* ***plausibly*** *explained or "reconstructed" by visual data from the second image?* Failing to perform this reconstruction implies an absence of information and thus, a disappearing object. Occlusion detection can then be defined as a spatiotemporal reconstruction problem.

For this problem to be well-posed there are two main issues. First, quality of the visual reconstruction has to be assessed with respect to a reference information. Incidentally, *the true reconstruction is available here*, and is precisely the same first image! Note that we reason directly on the quality of the reconstruction (with known reference) rather than indirectly on the quality of motion (unknown field). Sec-

[1] In the sense of the projection of the true physical motion onto the image plane.

ond, the reconstruction should be *plausible*, meaning that the way we pull information from the second image to reconstruct the first, must be consistent with apparent scene changes. Without the latter condition, one can get away with physically improbable information flows.

We start by focusing first aspect of the problem, that is, how to construct an occlusion criterion able to reason on the basis of image reconstruction. The criterion itself assumes that the reconstruction is plausible and consistent. A plausible reconstruction from a pair of images can be generated given a plausible correspondence motion field between them. This defines a natural way of pulling information from one image towards the other. A non-plausible reconstruction would be one that propagates color information between points that do not physically relate. Such a reconstruction is not necessarily useless, as many problems in image processing are not interested in the interpretation of the correspondence itself, like motion-compensated image compression, nearest neighbor search (*e.g.* [6]) or video denoising.

This puts forward the fact that motion indeed intervenes, leading us to the second aspect of the problem. We propose a complete framework for occlusion map estimation which exploits dynamic idiosyncrasies of the scene of interest. Indeed, the plausibility of the reconstruction (loosely, how probable it is) *does not demand accuracy in motion estimation nor a hard decision on which is the optimal correspondence vector*.

4. Proposed occlusion detection

We start by defining a reconstruction-based criterion for independently detecting occluded pixels (Section 4.1), assuming the knowledge of a correspondence map. In Section 4.2, we then leverage this new local model within an image-wise formulation of occlusion detection, where instrumental correspondences are obtained from a collection of suitable parametric motion models. No accurate optical flow is thus required while searching the best binary occlusion labeling over the image.

4.1. A reconstruction-based criterion

Let us define two functions $\zeta(I)$ and $\eta(I';I,\mathbf{w})$ that provide two different *reconstructions* of the same image I, either from itself (intra-image reconstruction) or from another related image I', a correspondence field between the two, $\mathbf{w}$, being given (inter-image reconstruction under correspondence guidance). Given a pair (I_1, I_2) of successive video frames and some correspondence field $\mathbf{w}$ from the first image to the second one, we shall denote in short $\zeta_1 = \zeta(I_1)$ and $\eta_{1,2} = \eta(I_2; I_1, \mathbf{w})$.

In essence, $\eta_{1,2} = \big\{\eta_{1,2}(\mathbf{x})\big\}_{\mathbf{x}\in\Omega}$ conveys the appearance of image I_1 that is retained by the second image. As such, under the true motion field, $\eta_{1,2}$ is expected to deviate towards the appearance of I_2 for all the occluded points. This particular behavior is clearly visible in Fig. 3, where segments with largest motions appear doubled in a stroboscopic-like effect. On the other hand, ζ_1 captures the intrinsic appearance of I_1 revisited from its own perspective for every $\mathbf{x}$ on the image grid Ω.

This means that, if the two functions are defined in a suitable way, one could deduct whether a pixel at location $\mathbf{x}$ in the first image of the pair is visible or not in the second one by comparing ζ_1 and $\eta_{1,2}$ around this location. Experiments showed that this comparison is better conducted in an asymmetric way whereby a local color model g (defined later in Eq. 6) is fitted to ζ_1 in the neighborhood of $\mathbf{x}$ and used to assess the likelihood of $\eta_{1,2}(\mathbf{x})$ under visibility hypothesis.

Reasoning independently at the pixel level for now, point $\mathbf{x}$ will be considered as not visible in the second image if:

$$-\ln g\big(\eta_{1,2}(\mathbf{x})\big) > \varepsilon_v, \tag{2}$$

where ε_v is a conveniently chosen threshold, and g an exponential density function.

The function ζ aims mostly at "simplifying" the first input image I_1 such that robust comparisons can be conducted later on. Yet, it is important to preserve the structure of the input image. A natural choice for this function is the classic bilateral filter [34]:[2]

$$\zeta_1(\mathbf{x}) = \frac{1}{Z(\mathbf{x};I_1)} \sum_{\mathbf{y}\in N_{\mathbf{x}}} \alpha(\mathbf{x},\mathbf{y};I_1) I_1(\mathbf{y}), \tag{3}$$

where $N_{\mathbf{x}}$ is a square window centered at $\mathbf{x}$, $Z(\mathbf{x};I_1) = \sum_{\mathbf{y}\in N_{\mathbf{x}}} \alpha(\mathbf{x},\mathbf{y};I_1)$ is a normalization factor and the weighting function α depends on both appearance and spatial proximity within pixel pair:

$$\alpha(\mathbf{x},\mathbf{y};I_1) = f_a(\|I_1(\mathbf{y}) - I_1(\mathbf{x})\|) f_s(\|\mathbf{y}-\mathbf{x}\|), \tag{4}$$

with f_a and f_s being Gaussian kernels.

In a similar fashion, $\eta_{1,2}$ will form a structure-preserving reconstruction of the first image I_1 but, this time, using colors from the second image I_2 under the guidance of correspondence map $\mathbf{w}$:

$$\eta_{1,2}(\mathbf{x}) = \frac{1}{Z(\mathbf{x};I_1)} \sum_{\mathbf{y}\in N_{\mathbf{x}}} \alpha(\mathbf{x},\mathbf{y};I_1) I_2\big(\mathbf{y}+\mathbf{w}(\mathbf{y})\big). \tag{5}$$

This can be seen as a "displaced cross-bilateral filter", that is, the cross-bilateral filtering [19] of a warped image. Note that, as previously stated, we make use of a correspondence map only as a tool to find a valid reconstruction of I_1.

It is important that the two reconstruction functions share the same filter weights.[3] This way, $\eta_{1,2}$ captures as much as possible the local structure of I_1 and both reconstructions are comparable pixel-wise. Intentionally, this

[2] Any other discontinuity-preserving image filter could be used, provided it possesses a guided version.

[3] In particular, we have the desirable property $\zeta(I) = \eta(I; I, \mathbf{0})$.

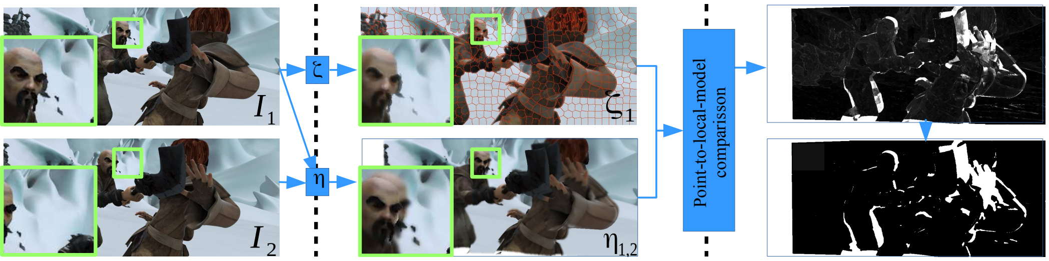

Figure 3. **Pipeline for proposed occlusion detection criterion**. Given successive images I_1 and I_2, functions ζ and η generate two "reconstructions" ζ_1 and $\eta_{1,2}$ of I_1. The second reconstruction is obtained from I_2, under the guidance of a given motion field and of I_1 (to preserve the structures of it). An arbitrary window of an occluded zone is zoomed-in for inspection. The likelihood of the color at each pixel of $\eta_{1,2}$ is evaluated under the corresponding local model extracted from ζ_1 at the super-pixel level (*point-to-local-color-model comparison*). This provides a soft-occlusion map that can be either thresholded pixel-wise to obtain a binary occlusion map, or embedded in the unaries of a joint labeling cost function (not shown here).

structure-mimicking behavior is not favorable for reconstructing points that are visible only in the first image, *i.e.*, occluded points.

The next step in the procedure consists in assessing whether a point $\mathbf{x}$ is occluded or not based on the criterion defined in (2). In order to conduct this step in practice, we propose here to over-segment the first reconstructed image into homogeneous segments where meaningful local color models can be estimated. In our experiments, these homogeneous segments are SLIC superpixels [1], and a Gaussian Mixture Model (GMM) of color is extracted for each of them, defining the density g in (2).

Image ζ_1 is segmented into J super-pixels and the j-th one, $S_j \subset \Omega$, is equipped with the mixture:

$$g_j = \sum_{k=1}^{K_j} \pi_k^j \mathcal{G}(\mu_k^j, \Sigma_k^j), \tag{6}$$

where π_k^j, μ_k^j and Σ_k^j are respectively the weight, the mean and the covariance matrix of the k-th component of the mixture. These GMMs provide good local models of ζ_1 in the sense that one can assume that

$$\forall \mathbf{x} \in \Omega,\ \zeta_1(\mathbf{x}) \sim g_{s(\mathbf{x})}, \tag{7}$$

where $s(\mathbf{x}) \in [\![1, J]\!]$ is the index of super-pixel containing $\mathbf{x}$. This should also hold for $\eta_{1,2}(\mathbf{x})$ provided the point is not occluded in second image. The novel reconstruction-based test for occlusion detection at the pixel level (2) finally reads:

$$\mathbf{x} \text{ occluded if } -\ln g_{s(\mathbf{x})}\big(\eta_{1,2}(\mathbf{x})\big) > \varepsilon_v. \tag{8}$$

Contrary to DFD-based test (1), this one is not based on a point-to-point comparison but on a point-to-local-model one. We shall demonstrate experimentally that it is a more powerful alternative. Yet, as DFD-based test, it still assumes that a correspondence map is available to produce the temporal image reconstruction $\eta_{1,2} = \eta(I_2; I_1, \mathbf{w})$. Next, we explain how to use this novel modeling over the whole image without depending on a single, accurate motion field.

4.2. From motion models to occlusions without stopping by optical flow

We propose a method for detecting occlusions which uses the image-reconstruction reasoning explained above. In order to be agnostic to optical flow computation, we propose to rely on a collection of *motion models* that spans the various dynamics of the scene and thus enables plausible image reconstructions. As classically exploited in video segmentation and analysis, the apparent motion at work in natural dynamic scenes can often be decomposed into a set of low-complexity models, typically region-wise affine models. Such a paradigm recently proved useful also to estimate dense optical flows [43]. In our case, such models will provide candidate correspondences at each pixel, leading to a discrete labeling problem intertwined with the main one of occlusion detection.

We start by computing a set $\mathcal{W} = \{\mathbf{w}_k,\ k = 1 \cdots K\}$ of K parametric motion models that are relevant to different sub-regions of the scene. We extract a large number K of overlapping windows of different sizes, starting with a window encompassing the full image support, and subsequently reducing the size by a half and changing position of the windows with a fixed overlap factor of 50%, covering the whole image for every window size. For the image size of the Sintel dataset, with four levels of window sizes we obtain, for instance, $K = 115$ windows.

For each window, we robustly estimate a parametric warp that captures at best the motion of corresponding scene fragment. Several classic techniques can be used to this end. In our experiments, we combine semi-dense matching, to handle large displacements, with robust affine motion estimation. We first extract point matches between images I_1 and I_2 with *DeepMatching*[4] [24] and fit an initial affine motion to the matches originated from the window of interest. These models are then refined with *Motion2D*[5], an M-estimator relying on all support intensities [20].

[4] `http://lear.inrialpes.fr/src/deepmatching/`
[5] `http://www.irisa.fr/vista/Motion2D/`

This multi-window motion estimation approach arguably provides a partially redundant collection of motion models, but it is simple and it circumvents in particular the intricate problem of motion segmentation. From the set $\mathcal{W}$ of K parametric motion models thus obtained, we want to exploit the most plausible at each pixel to achieve occlusion detection. Observe that with this procedure, pixel-wise occlusion modeling is not tied to a single, accurate, dense optical flow, but rather to a region-wise characterization of the scene dynamics.

The task to solve is now the one of jointly selecting a motion model and deciding on visibility for each pixel. We pose it as an energy minimization problem with respect to a motion model labeling $M = \big\{m(\mathbf{x})\big\}_{\mathbf{x}\in\Omega} \in [\![1, K]\!]^{\Omega}$ and an occlusion map $O = \big\{o(\mathbf{x})\big\}_{\mathbf{x}\in\Omega} \in \{0,1\}^{\Omega}$, where 1 means occluded point. For pixel location $\mathbf{x}$, label $m(\mathbf{x})$ indicates which one of the available parametric motion models is relevant, while $o(\mathbf{x})$ establishes if there is an occlusion or not. For $m(\mathbf{x}) = k$, the associated inter-image reconstruction (5) is denoted $\eta^k_{1,2}(\mathbf{x})$. The joint energy to minimize is defined as:

$$E(O,M) = \sum_{\mathbf{x}\in\Omega} \phi_{\mathbf{x}}(o(\mathbf{x}), m(\mathbf{x})) + \mathrm{DL}(M) + \sum_{\mathbf{x}\sim\mathbf{y}} \big(\psi^m_{\mathbf{x},\mathbf{y}}(m(\mathbf{x}), m(\mathbf{y})) + \psi^o_{\mathbf{x},\mathbf{y}}(o(\mathbf{x}), o(\mathbf{y}))\big) \tag{9}$$

where the second sum is taken over all pairs of neighboring pixels. The unary potential reads

$$\phi_{\mathbf{x}}(0,k) = -\ln g_{s(\mathbf{x})}(\eta^k_{1,2}(\mathbf{x})),\ \phi_{\mathbf{x}}(1,k) = \alpha_v, \tag{10}$$

where $\alpha_v > 0$ is the cost of labeling a single pixel as occluded. It is related to ε_v in pixel-wise test (8). This data term is thus not based on point-wise displaced frame differences as classically done, but on reconstruction-based local modeling. Although this modeling effectively penalizes motions that do not preserve color information up to the precision of the local model, this data-term, as seen from the unknown motion point-of-view, would not be suitable to yield an accurate pixel-wise motion estimation. Again, this is not the intention anyway, the sole aim being to reason locally on as plausible as possible inter-image reconstructions. This is further analyzed in Section 5.

Since each motion label k corresponds to a specific image window, one could restrict the labeling of pixel $\mathbf{x}$ according to the windows it belongs to. We found, however, that this restriction can cause block-like artifacts in the label assignment, with subsequent damage to final occlusion labeling. To capture motion model locality in a less drastic way, we propose instead to double $\phi_{\mathbf{x}}(0,k)$ (unary potential for visible points) for motion models k stemming from windows $\mathbf{x}$ does not belong to.

The binary potentials share a similar form of contrast-sensitive smoothing:

$$\psi^a_{\mathbf{x},\mathbf{y}}(k,k') = \lambda_a \exp\left(-\beta_a \left\|I_1(\mathbf{x}) - I_1(\mathbf{y})\right\|\right)[k \neq k'] \tag{11}$$

with $a \in \{“o”, “m”\}$, where $[\cdot]$ is Iverson bracket and λ_o, λ_m are positive parameters.

Finally, the global motion label cost $\mathrm{DL}(M)$ penalizes the complexity of the labeling through its “description length”, i.e., the number of labels effectively used:

$$\mathrm{DL}(M) = \lambda_c \sharp\{k : \exists \mathbf{x} \in \Omega, m(\mathbf{x}) = k\}. \tag{12}$$

The occlusion map O and, as a by-product, the motion-model label map M, are obtained by minimizing $E(O,M)$ (Eq. 9) with a block coordinate descent in an alternate way. Given occlusion assignment O, minimizing $E(O,M)$ w.r.t. M only amounts to minimizing

$$\sum_{\mathbf{x}\in\Omega} \phi_{\mathbf{x}}(o(\mathbf{x}), m(\mathbf{x})) + \sum_{\mathbf{x}\sim\mathbf{y}} \psi^m_{\mathbf{x},\mathbf{y}}(m(\mathbf{x}), m(\mathbf{y})) + \mathrm{DL}(M). \tag{13}$$

This can be done approximately by using α-expansions with the method of [11] in order to handle the global label cost term. Subsequently, for a given motion model label map, the occlusion map can be recovered by minimizing w.r.t. O the following function:

$$\sum_{\mathbf{x}\in\Omega} \phi_{\mathbf{x}}(o(\mathbf{x}), m(\mathbf{x})) + \sum_{\mathbf{x}\sim\mathbf{y}} \psi^o_{\mathbf{x},\mathbf{y}}(o(\mathbf{x}), o(\mathbf{y})), \tag{14}$$

with graph-cuts [8]. The occlusion and label maps are alternatively updated for a small number of iterations. Recall that this process is not oriented at optical flow recovery, but at selecting plausible motion models.

5. Experimental results

For the quantitative evaluations reported in this section, we rely on the MPI Sintel dataset[6] [10]. This dataset comprises 69 sequences from the open-source CGI movie Sintel[7] for which ground-truth optical flows and occlusion maps have been computed from the known 3D dynamic structure of the scenes.

5.1. Evaluation of the occlusion criterion

Let us first demonstrate the value of the reconstruction-based criterion introduced in Section 4.1 by performing pixel-wise occlusion detection under the guidance of three different correspondence fields: (1) The true optical flow, as accessible in MPI Sintel sequences; (2) The true flow contaminated by independent additive Gaussian noise of standard deviation 2.5; (3) The flow estimated with *DeepFlow*[8][38], a state-of-art optic flow estimator which does not handle occlusions but solves for long displacements.

[6] http://sintel.is.tue.mpg.de/downloads

[7] https://durian.blender.org/download/

[8] http://lear.inrialpes.fr/src/deepflow/

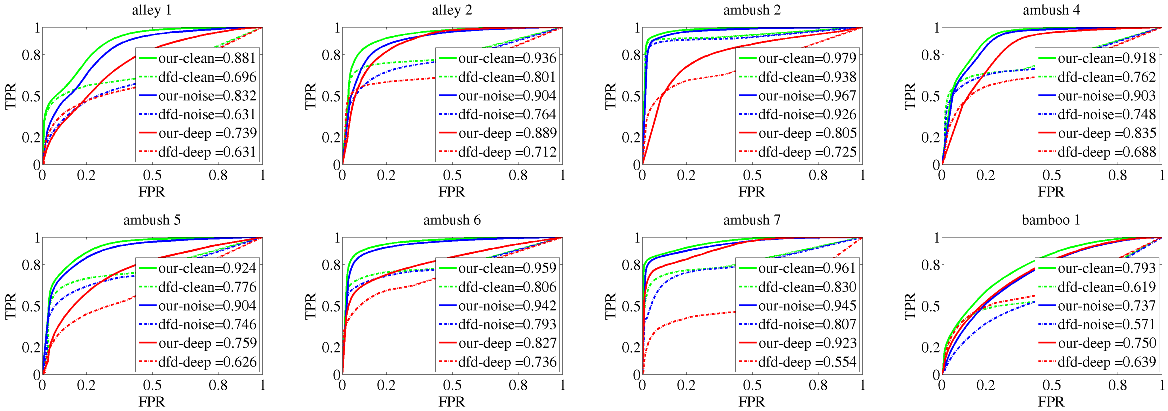

Figure 4. **Quantitative comparison of proposed reconstruction-based criterion and classic DFD-based criterion**. Occlusion detection ROC curves and associated AUC on eight sequences of the *MPI Sintel* dataset, obtained by varying the threshold of proposed criterion (solid lines) and of DFD criterion (dashed lines). Colors indicate the origin of the motion field: true optical flow (green), true optical flow contaminated by Gaussian noise (blue), *DeepFlow* estimate (red).

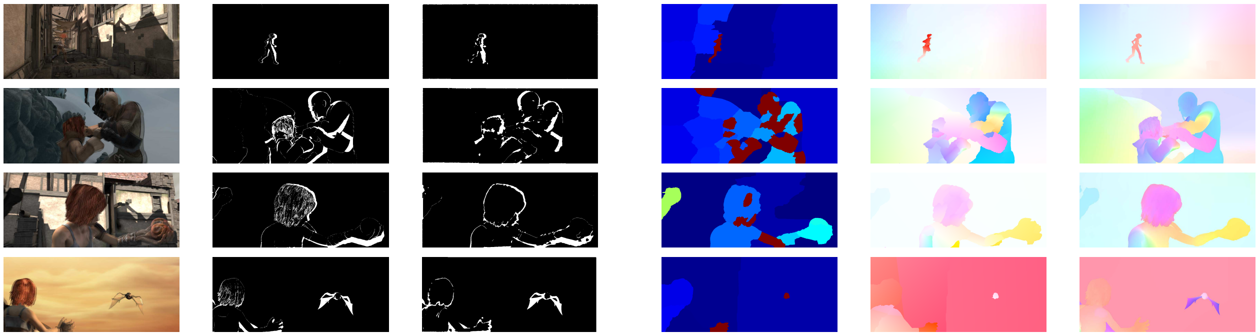

Figure 5. **Qualitative results of our occlusion detection method on several MPI Sintel scenes**. From left to right: Average of the two input frames; Occlusion ground-truth; Occlusion map with proposed method; Final motion model labeling, where each color loosely represents the size and position of the window linked to the selected motion model through a jet-map colorization (Big to small and from upper-left to bottom-right); Optical flow obtained by evaluating the selected motion model at each pixel (with classic hue-saturation color coding); And true optical flow with same color coding.

For all the experiments, we fixed the number of super-pixels J to 700. We found experimentally that having only 2 components for each *GMM* provides good results in general. Furthermore, the standard deviation of the Gaussian kernels was set to 1.0, with a window size of 5 pixels. These parameters are related to the local color model variability within the superpixel supports.

Comparisons are conducted against DFD-based detection, which is at the heart of approaches based on analyzing violations of the brightness constancy assumption [4, 40]. For eight sequences of the dataset, we report in Fig. 4 occlusion detection ROC curves (true-positive-rate (TPR) vs. false-positive-rate (FPR)) and area under the ROC curves (AUC), both varying the threshold ε_c in the rule (1) and varying ε_v in our own framework (8). It can be seen that our reconstruction-based criterion has more discriminative power than classic DFD criterion in the prospect of occlusion detection. In all regimes and on all the sequences the former outperforms the latter, often by a substantial margin. More interestingly, for most of the sequences, *i.e.*, “alley_1”, “alley_2”, “ambush_4”, “ambush_5”, “ambush_7” and “bamboo_1”, the decrease in occlusion detection performance due to flow inaccuracies (with noisy ground-truth or *DeepFlow* estimate) is less significant for our method. Remarkably, when used together with *DeepFlow*, our criterion outperforms on several sequences the DFD criterion based on true motion field.

5.2. Full system evaluation

We now turn to the complete minimization-based method introduced in Section 4.2. We compare it on MPI Sintel sequences to several recent approaches: (1) a learning-based occlusion detection algorithm, which uses a non trivial ensemble of hand-designed features [14], including forward-backward flow inconsistencies for several optical flow methods; (2) a method based on layer assignment and depth ordering [30]; (3) a method that leverages reasoning on local layers relationships [29]; (4) a sparse occlusion detection method that relies on departures from the optical flow color constancy assumption [4].

In the experiments, we limit the number of iterations of our alternate minimization method to 2, as it converges

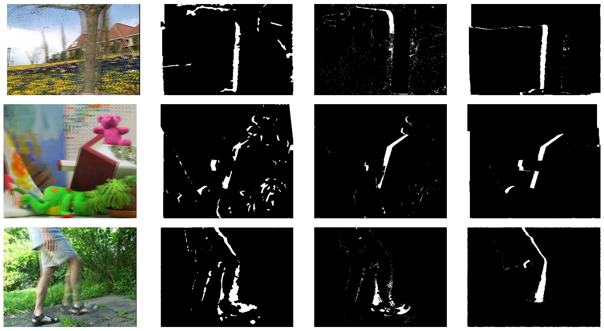

Figure 6. **Qualitative comparison with state-of-the-art on real images**. From left to right: Average of the two input frames, final occlusion map with proposed method, results of [4], and of [30].

quickly. Furthermore, we set $\lambda_o = 20.0$, $\lambda_m = 33.3$, $\beta_o = 0.1$, $\beta_m = 0.2$, and $\lambda_c = 10^3$ by hand-tuning. Setting critical parameter α_v is discussed below.

Table 1 summarizes results by the average F-score computed over all 69 ground-truth MPI Sintel sequences, using two ways of setting the main occlusion parameter for each method, *e.g.*, α_v for our method. In a first set of experiments ("Global 69" column), it is manually set at once for all sequences. For [29] and [14], the corresponding parameters are set to 0.5, as reported by [29]; for our method α_v is set to 10; finally, for [4][9] and [30][10] we set the parameters to the values proposed by the respective authors. In a second round ("Oracle 69" column), we tuned the key parameter of each method so as to maximize the F-scores. As it can be appreciated, our reconstruction-based method outperforms all the other reported methods in both settings. Finally, we also present some results (as available) by tuning the parameters to maximize the F-scores only in the 23 training sequences on the final rendering pass of the Sintel dataset ("Oracle Final" column).

We provide in Fig. 5 several samples of our results for visual inspection. Despite the complexity of some of these scenes, occlusion maps of good quality are obtained. In particular, extended occlusions produced by large displacements are very well captured, while retaining some details of smaller occluded regions. It is also interesting to examine associated motion model labelings $M = \{\mathbf{m}(\mathbf{x})\}_{\mathbf{x}\in\Omega}$ and associated motion flows $\{\mathbf{w}^{\mathbf{m}(\mathbf{x})}(\mathbf{x})\}_{\mathbf{x}\in\Omega}$ (Fig. 5 e-f). While motion labels define segments that relate well to actual moving regions (*e.g.* the arm of the woman in the fourth sequence or the whole person in the first one), the flows are not very accurate for all the sequences. This is not surprising since source motion models have been estimated beforehand over arbitrary image windows. Motion model selection with no further processing cannot produce accurate optical flows. This reveals, as already highlighted in Section 4.2, that the distinctive nature of our framework is to focus on good reconstruction-based modeling of occlusion, with inference requiring only plausible correspondences, not accurate per-pixel optical flow. The local color modeling, however, carries a few problems that somewhat limit the performance of our method. For example, GMMs easily overfit in textureless regions, causing false positives under slight temporal illumination variations.

Table 1. **Quantitative comparisons on MPI Sintel dataset**. For each method, the average F-score (the higher, the better) is computed with two different ways of setting the main detection parameter over all the available training sequences and rendering passes. We also present results on the Final rendering pass.

Detection method	Oracle 69	Global 69	Oracle Final
Learning [14]	0.535	0.448	-
Depth order [30]	0.465	0.449	0.398
Local layers [29]	0.474	0.376	-
Sparse method[4]	0.310	0.259	0.258
Ours	**0.550**	**0.540**	**0.491**

Regarding execution times, for a sample image pair of size 436 × 1024, our full method takes about 1212 s (29 s for the multi-window motion estimation and 1183 s for the energy minimization) on an Intel i7-3540M CPU @ 3.00GHz. In comparison, [4] takes 1601 s, while [30] takes 2201 s on the same machine.

Finally, we show in Fig. 6 additional comparative results on real-world image sequences. Qualitative assessment of these results further confirm the merit of our approach: more accurate occlusion maps are produced compared to [4] and [30].

6. Concluding remarks

We have introduced a new approach to detect occlusions that occur from one frame to another in a video sequence. Departing from classic approaches that tightly link this task to the accurate computation of optical flow, we propose a local spatio-temporal reconstruction model that only requires the knowledge of a plausible motion. Given a collection of parametric motion models simply extracted from the scene at hand, we show how this local reconstruction model can be harnessed within a global energy function to deliver high quality occlusion maps.

Quantitative experiments yielded two important findings: (1) Proposed reconstruction-based modeling provides a detection criterion at the pixel level that is a powerful alternative to classic criterion based on intensity inconsistency along the flow. Even when exploiting the true flow, later criteria perform less well that ours with the output of an off-the-shelf flow estimator. (2) Our complete energy-based framework consistently outperforms state-of-the-art approaches on MPI Sintel dataset. More generally, we qualitatively observed the ability of our framework to produce good quality occlusion detection maps, even in scenes comprising large occluded regions and complex motions.

[9] http://vision.ucla.edu/~ayvaci/spaocc.html

[10] We thank authors for providing us with their source code.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012*, 34(11):2274–2282.

[2] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sánchez. Symmetrical dense optical flow estimation with occlusions detection. *Int. J. of Computer Vision. 2007*, 75(3):371–385.

[3] N. Apostoloff and A. Fitzgibbon. Learning spatiotemporal t-junctions for occlusion detection. In *CVPR 2005*.

[4] A. Ayvaci, M. Raptis, and S. Soatto. Sparse occlusion detection with optical flow. *Int. J. of Computer Vision. 2012.*, 97(3):322–338.

[5] C. Ballester, L. Garrido, V. Lazcano, and V. Caselles. A TV-L1 optical flow method with occlusion detection. *Pattern Recognition*, 7476:31–40, 2012.

[6] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (TOG)*, 28(3):24, 2009.

[7] R. Ben-Ari and N. Sochen. Variational stereo vision with sharp discontinuities and occlusion handling. In *ICCV 2007*.

[8] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[9] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2003*, 25(8):993–1008.

[10] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV 2012*.

[11] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. *Int. J. of Computer Vision. 2012*, 96(1):1–27.

[12] V. Estellers and S. Soatto. Detecting occlusions as an inverse problem. *Journal of Mathematical Imaging and Vision. 2015.*, pages 1–18.

[13] D. Fortun, P. Bouthemy, and C. Kervrann. Aggregation of local parametric candidates with exemplar-based occlusion handling for optical flow. *Computer Vision and Image Understanding. 2016.*, 145:81–94.

[14] A. Humayun, O. Mac Aodha, and G. J. Brostow. Learning to find occlusion regions. In *CVPR 2011*.

[15] S. Ince and J. Konrad. Occlusion-aware optical flow estimation. *IEEE Transactions on Image Processing*, 17(8):1443–1451, 2008.

[16] N. Jacobson, Y. Freund, and T. Q. Nguyen. An online learning approach to occlusion boundary detection. *IEEE Transactions on Image Processing*, 21(1):252–261, 2012.

[17] S. B. Kang, R. Szeliski, and J. Chai. Handling occlusions in dense multi-view stereo. In *CVPR 2001*.

[18] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *ICCV 2001*.

[19] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele. Joint bilateral upsampling. In *ACM Transactions on Graphics (TOG)*, volume 26, page 96, 2007.

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

[21] J. Pan and B. Hu. Robust occlusion handling in object tracking. In *CVPR 2007*.

[22] M. Proesmans, L. Van Gool, E. Pauwels, and A. Oosterlinck. Determination of optical flow and its discontinuities using non-linear diffusion. In *ECCV 1994*.

[23] S. H. Raza, A. Humayun, I. Essa, M. Grundmann, and D. Anderson. Finding temporally consistent occlusion boundaries in videos using geometric context. In *WACV 2015*.

[24] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. DeepMatching: Hierarchical deformable dense matching. *Int. J. of Computer Vision. Submitted in 2015.*

[25] M. E. Sargin, L. Bertelli, B. S. Manjunath, and K. Rose. Probabilistic occlusion boundary detection on spatio-temporal lattices. In *ICCV 2009*, pages 560–567.

[26] P. Smith, T. Drummond, and R. Cipolla. Layered motion segmentation and depth ordering by tracking edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2004*, 26(4):479–494.

[27] A. N. Stein and M. Hebert. Occlusion boundaries from motion: Low-level detection and mid-level reasoning. *Int. J. of Computer Vision. 2009*, 82(3):325–357.

[28] C. Strecha, R. Fransens, and L. Van Gool. A probabilistic approach to large displacement optical flow and occlusion detection. In *Statistical methods in video processing*, pages 71–82. 2004.

[29] D. Sun, C. Liu, and H. Pfister. Local layering for joint motion estimation and occlusion detection. In *CVPR 2014*.

[30] D. Sun, E. B. Sudderth, and M. J. Black. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. In *NIPS 2010*.

[31] D. Sun, E. B. Sudderth, and M. J. Black. Layered segmentation and optical flow estimation over time. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1768–1775. IEEE, 2012.

[32] J. Sun, Y. Li, S. B. Kang, and H.-Y. Shum. Symmetric stereo matching for occlusion handling. In *CVPR 2005*.

[33] P. Sundberg, T. Brox, M. Maire, P. Arbeláez, and J. Malik. Occlusion boundary detection and figure/ground assignment from optical flow. In *CVPR 2011*.

[34] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV 1998*.

[35] A. Verri and T. Poggio. Motion field and optical flow: Qualitative properties. *Pattern Analysis and Machine Intelligence, IEEE Transactions on. 1989.*, 11(5):490–498.

[36] Y. Wang and G. Mori. Multiple tree models for occlusion and spatial constraints in human pose estimation. In *ECCV 2008*.

[37] D. Weinland, M. Özuysal, and P. Fua. Making action recognition robust to occlusions and viewpoint changes. In *ECCV 2010*.

[38] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *ICCV 2013*.

[39] B. Wu, R. Nevatia, and Y. Li. Segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. In *CVPR 2008*.

[40] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. In *ECCV 2006*.

[41] L. Xu, J. Chen, and J. Jia. A segmentation based variational model for accurate optical flow estimation. In *ECCV 2008*.

[42] L. Xu, J. Jia, and Y. Matsushita. Motion detail preserving optical flow estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012*, 34(9):1744–1757, 2012.

[43] J. Yang and H. Li. Dense, accurate optical flow estimation with piecewise parametric model. In *CVPR 2015*.

[44] A. Yilmaz, X. Li, and M. Shah. Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2004*, 26(11):1531–1536.