Self-Occlusion and Disocclusion in Causal Video Object Segmentation

Yanchao Yang\textsuperscript{1}, Ganesh Sundaramoorthi\textsuperscript{2}, and Stefano Soatto\textsuperscript{1}

\textsuperscript{1}University of California, Los Angeles, USA \hspace{1em} \textsuperscript{2}King Abdullah University of Science & Technology (KAUST), Saudi Arabia

\texttt{yyc8912@g.ucla.edu, ganesh.sundaramoorthi@kaust.edu.sa, soatto@ucla.edu}

Abstract

We propose a method to detect disocclusion in video sequences of three-dimensional scenes and to partition the disoccluded regions into objects, defined by coherent deformation corresponding to surfaces in the scene. Our method infers deformation fields that are piecewise smooth by construction without the need for an explicit regularizer and the associated choice of weight. It then partitions the disoccluded region and groups its components with objects by leveraging on the complementarity of motion and appearance cues: Where appearance changes within an object, motion can usually be reliably inferred and used for grouping. Where appearance is close to constant, it can be used for grouping directly. We integrate both cues in an energy minimization framework, incorporate prior assumptions explicitly into the energy, and propose a numerical scheme.

1. Introduction

Persistent tracking of three-dimensional (3D) objects in video presents long-standing challenges unless they are flat \cite{33}, or the video is short \cite{25}. As surfaces move in 3D relative to the viewer, previously unseen portions of the scene become visible and will have to be attributed to different objects to maintain tracking. Such disocclusion phenomena are the focus of our investigation.

Consider a camera rotating around a box in Fig 1: Both the occluded and disoccluded regions involve portions of different objects, in this case just the box and the “background.” Occlusions have been addressed by \cite{29, 1}. We focus on disocclusions, by determining the disoccluded area (Sect. 2), partitioning it and grouping each portion with an object (Sect. 3).

Grouping unseen portions of the scene into different objects requires prior assumptions on their properties. One could assume that the “appearance” or “texture” of objects is homogeneous (\textit{i.e.}, their reflectance exhibits spatially stationary statistics) and leverage on the similarity of image color histograms to partition and group disoccluded regions. However, this assumption often fails, as in Fig. 1. Alternatively, one could assume that the “apparent motion” of objects is homogeneous (\textit{i.e.}, the deformation undergone by the image domain is smooth within objects, and discontinuous across). However, when objects exhibit “textureless” surfaces (\textit{i.e.}, constant reflectance), such a deformation is undetermined, and cannot be used for grouping.

Fortunately, motion and appearance cues are complementary: When one fails to be informative, the other may be. Leveraging such complementarity is central to this paper. When the disoccluded region exhibits complex appearance, motion can be reliably inferred and exploited for grouping. Otherwise, when the disoccluded region is textureless, photometric statistics are spatially homogeneous and can be reliably used for grouping. Of course, both cues can fail if an object has piecewise constant appearance, and the transition happens right at the disocclusion (Fig. 2). However, these are accidental phenomena that do not persist in long temporal sequences.

For us, objects are layouts of piecewise smooth and smoothly deforming surfaces in 3D supporting Lambertian reflection seen under constant illumination throughout a video sequence. There can be multiple objects moving independently, in addition to viewer (or equivalently background) motion. Under these assumptions, the domain of a video image of a scene can be partitioned into two types of regions: Those that are \textit{co-visible}, that under the stated assumptions are a smooth deformation of regions in the pre-
vious frame, and those that are disoccluded, i.e., whose pre-image under perspective projection is a portion of a surface that was not visible in the previous frame(s). In addition, occluded regions are subsets of a region that, in the previous frame, was occupied by an object different from the current one. These have been addressed by others [1].

Disoccluded regions in a video are the occluded regions in the video played backwards. Because we eventually aim at real-time closed-loop operation, we wish to process the data causally. Furthermore, parts of objects can appear in a frame and disappear in the next, a case which forward-backward sweeps would not address (Sect. 4). With an abuse of nomenclature, we refer to “objects” as both the connected surfaces in 3D, and the subsets of the (2D) image domain where they project.

Contributions: To detect disocclusions, we extend the Sobolev framework of [38] to multiple objects (Sect. 2). This framework naturally encompasses coarse-to-fine deformation inference without an explicit regularizer and the associated weighting constant. To partition and group disoccluded regions to various objects, we leverage on the complementarity of motion and appearance cues by introducing a novel data term that encompasses both (Sect. 3). We derive an efficient numerical scheme and test it against competing methods on benchmark datasets (Sect. 4).

1.1. Related work

Persistent object tracking in video touches upon a large body of work in video segmentation (e.g., [14, 19, 36, 16]), tracking (e.g., [35, 3, 12, 20]), optical flow (e.g., [15, 6, 7, 39, 26]), and motion segmentation (e.g., [33, 27]). In dealing with visibility phenomena, our work relates to occlusion detection. There is a literature on detecting occluding boundaries from static images or short-baseline video (see [29] and references therein). Since we tackle persistent tracking, we do not discuss this further. Our work is related to [1] that partitions the image domain into (flat) layers like [33], but in a convex optimization setting after relaxing the ℓ₀ norm to ℓ₁. We detect occlusions without the need for such a relaxation and without the need for regularization of the deformation field, which can cause over-smoothing in some regions, and under-smoothing in others. Instead, following [38] we employ a Sobolev approach [28] (see also [9, 4]) to infer deformation fields that are by construction smooth in a naturally coarse-to-fine manner. On a short time-scale, such deformation fields are related to optical flow, which we do not review here, except for when the flow is partitioned into regions, as in motion segmentation. There, the flow field is often assumed to be piecewise parametric. Here we allow each component to be a diffeomorphism to handle articulated and deforming objects without over-segmenting them. Other motion segmentation approaches perform clustering of optical flow, often non-causally [23, 14].

Although our goal is segmentation, our method produces diffeomorphic warps, and relates to diffeomorphic registration, e.g., [4, 32, 10]. We produce a piecewise diffeomorphism of the image rather than a global diffeomorphism as in [4, 32, 10], an assumption that breaks under (dis)occlusions. Also, our warp computation is parameter-free in contrast to [4, 32, 10].

Taylor et al. [30] perform layer segmentation in longer video sequences leveraging occlusion cues, but do not explicitly address the interplay of motion and intensity cues in disocclusion. Similarly, [27] performs layered segmentation by grouping. Only intensity cues are used for the disocclusion in [8, 38].

This work also relates to dense 3D reconstruction of geometry and photometry [18, 22, 37, 13, 17], since an explicit 3D reconstruction of the scene produces as a side effect a partition of the video into regions. However, it requires a static scene, and does not address deforming objects moving independently, which our work addresses.

2. Sobolev Warps and Occlusions

We seek to partition the domain D of a time-varying color image \( I_t : D \subset \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) for \( t = 1, 2, \ldots \), into a collection \( \{R_i\}_{i=1}^N \) of regions \( R_i \). We omit the time index hereafter for simplicity. These regions are also called “objects,” that move coherently, as defined next.

The (apparent) motion of each region \( R_i \), also referred to as a warp or a deformation, is defined in the domain of the image \( I_t \) as the map \( w_i : R_i \rightarrow D \) that transforms \( I_{t+1} \) back to \( I_t \). Assuming the scene is Lambertian, illumination is constant, and the image is corrupted by additive zero-mean Gaussian noise, the maximum-likelihood estimate of \( w_i \) is obtained by minimizing \( E_{\text{warp}}(w_i, O_i) \), given by

\[
E_{\text{warp}} = \int_{R_i \setminus O_i} |I_{t+1}(w_i(x)) - I_t(x)|^2 \, dx + \beta \int_{O_i} \, dx, \tag{1}
\]

where \( O_i \subset R_i \) is the (unknown) occluded region that is visible at time \( t \) but not at time \( t+1 \). Note that, although \( w_i \) is defined on all of \( R_i \), the data \( I_{t+1}, I_t \) only provides evidence of it in the co-visible region \( R_i \setminus O_i \). To avoid the trivial solution \( O_i = R_i \) and thus \( w_i \) undetermined, we put a penalty on the occluded area as in [1].

Eq. (1) is reminiscent of many optical flow estimation algorithms [15, 6, 7, 39], but there are important differences: First, each warp is restricted to a subset \( R_i \subset D \) with no compatibility condition or relation among the different warps. Second, there is no regularizer for the warps. Most motion segmentation or optical flow schemes either assume that each warp belongs to a (small-dimensional) parametric family such as the group of affine transformations, or impose a penalty on the (piecewise) smoothness of \( w_i \). Instead, we leverage on the Sobolev framework [28] to impose
regularity in a naturally coarse-to-fine framework, while allowing the warps to be arbitrary diffeomorphisms (smooth maps with a smooth inverse). So, rather than adding a regularizer for the warps in (1), we compute each warp as the integral of a smooth time-varying vector field that, at each instant, belongs to a Sobolev space. This allows us to efficiently optimize (1) without imposing global regularization, which may be too much for fine-scale objects, and too little for large ones.

Given the warp $w_i$, the optimal occlusion $O_i$ is

$$O_i = \{x \in R_i : |I_{t+1}(w_i(x)) - I_t(x)|^2 > \beta\}.$$  

(2)

Substituting the expression above into the energy, we obtain

$$E_{\text{warp}}(w_i) = \int_{R_i} \rho(I_{t+1}(w_i(x)) - I_t(x)) \, dx,$$

(3)

which now depends only on the warp $w_i$, and where

$$\rho(y) = |y|^2 \text{ for } |y|^2 < \beta \quad \text{and} \quad \rho(y) = \beta \text{ for } |y|^2 \geq \beta.$$  

(4)

With this, we can finally clarify the notion of “coherent motion” used to define the regions $R_i$: A region $R_i$ moves coherently if there is a warp $w_i$ that is smooth according to the Sobolev metric, that (locally) minimizes (3).

The gradient of $E_{\text{warp}}$, $G_i : w_i(R_i) \rightarrow \mathbb{R}^2$, with respect to the Sobolev metric has been computed by [38] and is

$$G_i(x) = \nabla_{\text{Sob}} E(w_i)(x) = \frac{1}{\alpha} \tilde{G}_i(x),$$

(5)

where $\alpha > 0$ is a parameter that will be eliminated below. $F_i : w_i(R_i) \rightarrow \mathbb{R}^2$ is

$$F_i = \nabla I_{t+1} \nabla \rho(I_{t+1} - I_t \circ w_i^{-1}) \det \nabla w_i^{-1},$$

(6)

$avg(F_i)$ is the average over $w_i(R)$, $\nabla$ is the vector of partials, and $\tilde{G}_i$ satisfies the partial differential equation (PDE):

$$\begin{cases} -\Delta \tilde{G}_i(x) = F_i(x) - avg(F_i) & x \in w_i(R_i) \\ \nabla \tilde{G}_i(x) \cdot N = 0 & x \in \partial w_i(R_i) \\ avg(\tilde{G}_i) = 0 \end{cases}$$

(7)

where $\Delta$ is the Laplacian, $N$ is normal to $\partial w_i(R_i)$, $\tilde{G}_i$ is the deformation, and $avg(F_i)$ is the translation.

To extend the framework to multiple regions, we extend each warp $w_i$ to the entire domain $D$ by imposing $\Delta \tilde{G}_i(x) = 0$ for $x \in D \setminus R_i$ and a Dirichlet condition on $\partial R_i$. The extension is continuous, but not differentiable across $R_i$.

Starting with the identity map $w_i(x) = x$, we deform it by the gradient descent (5) as follows. Define $\phi_i^{0,\tau} : D \rightarrow D$ and $\phi_i^{\tau,0} : D \rightarrow D$ as the evolving warp and its inverse where $\tau$ is an artificial time variable parameterizing the evolution. The inverse is needed to compute $F_i$. The evolution of the warps according to the gradient descent of $E_{\text{warp}}$ is

$$G_i^{\tau} = \nabla_{\text{Sob}} E_{\text{warp}}(\phi_i^{0,\tau}),$$

(8)

$$\partial_\tau \phi_i^{0,\tau}(x) = \nabla \phi_i^{\tau,0}(x) \cdot G_i^{\tau}(x),$$

(9)

$$\partial_\tau \phi_i^{\tau,0}(x) = -G_i^{\tau}(\phi_i^{0,\tau}(x))$$

(10)

for all $x \in D$. This gives a coarse-to-fine evolution. One can eliminate the parameter $\alpha$ by noting the independence of the deformation and translation components on $\alpha$ in (5). This gives Algorithm 1, which decreases the energy.

**Algorithm 1 Sobolev Warp Computation**

\[ \begin{align*} 
1: \text{Set } & \phi_i^{\tau,0}(x) = \phi_i^{0,\tau}(x) = x \text{ for } \tau = 0 \\
2: \text{repeat} & \\
3: \text{repeat} & \\
4: & \text{Let } \alpha \rightarrow \infty \text{ so } G_i^{\tau} = \text{avg}(F_i^{\tau}) \text{ is a translation} \\
5: & \text{Translate: Perform one iteration of (9)-(10)} \\
6: & \text{until } \text{avg}(F_i^{\tau}) = 0. \\
7: & \text{Deform: Do one iteration of (9)-(10) with } G_i^{\tau} = \tilde{G}_i^{\tau} \\
8: & \text{until } G_i^{\tau} = 0. \\
9: & \text{Set } w_i = \phi_i^{0,\tau_{\infty}} \text{ where } \tau_{\infty} \text{ is the convergence time} \\
\end{align*} \]

In Section 3.3, we will need to compute the occlusion so that it can be removed in the next frame. It can be computed at the end of the evolution as

$$O_i^{\infty} = \{x \in R_i : |I_{t+1} (\phi_i^{0,\tau_{\infty}}(x)) - I_t(x)|^2 > \beta\}.$$  

(11)

### 3. Causal Object Segmentation

If the motion of each region $R_i$ was reliably inferred, one could attempt to propagate forward the $R_i$ to segment the next frame. Unfortunately, regions that become *disoccluded* between $t$ and $t+1$ are not included in any of the $R_i$. While this is not a major problem if we are interested in only two adjacent frames, $t$ and $t+1$, as the area of the occluded/disoccluded regions is small, as time goes by the disocclusion typically grows. Thus, this phenomenon is hard to ignore when one considers long temporal sequences. The challenge becomes to assign the various components of the disocclusion to existing regions, or to spawn new ones. This is illustrated in Fig. 1: So long as the scene is populated by *non-flat* surfaces, multiple objects contribute to the disoccluded region.

We assume a partition into objects at time $t - 1$ and propagate it forward to time $t$. The disocclusion, i.e., the part of the domain $D$ not covered by the propagated segmentation, is initially assigned to regions based on estimated warps, and this is refined by minimizing the energy in Section 3.1.
Illustration of an error that arises in segmentation by grouping pixels only based on motion residuals. The object (dark yellow) moves to the left to occlude a portion of the background (dark green). Pixels in the occluded region are likely to be classified incorrectly in frame $t$ if only motion residuals are used since both residuals are large. When the background is constant in the occluded region and around it, classifying by residuals almost certainly leads to misclassifications.

### 3.1. Complementarity of Motion and Appearance

Of course both appearance and motion cues are obtained from image irradiance. What we mean by “cues” is bottom-up computation that leverages on the assumption of smooth spatial variation of image irradiance (appearance cues) versus smooth temporal variation of the same (motion cues).

To attribute disoccluded regions to any of the existing objects, we can leverage the photometric regularity and assign each segment to the object that has similar “texture” or motion. We favor the latter, as objects can have spatially-varying appearance, as in the cereal box in Fig. 1. This fails when the object and the background are textureless, as in Figure 2, or when they exhibit similar fine-scale texture. However, in this case grouping by appearance is straightforward. We leverage on this complementarity by exploiting preferentially motion regularity, consistent with our definition of objects, resorting to appearance regularity when the photometry is not suitable to reliably estimate motion.

#### Textless regions:

To leverage on this complementarity, we use the local standard deviation $\sigma_i(x)$ of $I_t$ in a neighborhood $B_{x,r'} \cap R_i$, where $B_{x,r'} = \{ y \in D : |x - y| \leq r' \}$ is the ball of radius $r'$ centered at point $x$. We can then define a measure of local constancy of any region local to a point $x$ as the minimum standard deviation over all regions that intersect the ball:

$$\sigma(x) = \min_{i, B_{x,r'} \cap R_i \neq \emptyset} \sigma_i(x).$$

(12)

Low values of $\sigma(x)$ indicate that the underlying color channels are not sufficiently exciting and therefore motion estimates can be expected to be unreliable.

#### Motion ambiguity function:

Grouping by residuals also should not be done when current warp residuals are large. Define the forward, backward and minimal residuals as

$$\text{Res}_f^i(x) = |I_{t+1}(w_f^i(x)) - I_t(x)|^2$$

(13)

$$\text{Res}_b^i(x) = |I_t(w_b^i(x)) - I_{t-1}(x)|^2$$

(14)

$$\text{Res}_i(x) = \min\{\text{Res}_f^i(x), \text{Res}_b^i(x)\}$$

(15)

where $w_f^i$ and $w_b^i$ are the current forward and backward warps of region $R_i$. The backward residual is used to remove some ambiguity in Fig. 2 as sometimes occluded pixels at time $t + 1$ are visible at time $t - 1$, and hence the backward motion may be reliable. The minimum of $\text{Res}_i$ over all regions that intersect with a ball around $x$,

$$\text{Res}_i(x) = \min_{i, B_{x,r'} \cap R_i \neq \emptyset} \text{Res}_i(x),$$

(16)

is small when motion cues are reliable. We define the motion ambiguity function, $\text{maf}: D \rightarrow \{0, 1\}$, which indicates whether motion cues are unreliable, as

$$\text{maf}(x) = \begin{cases} 1 & \text{if } \sigma(x) < k/r' \text{ or } \text{Res}(x) > \beta, \\ 0 & \text{otherwise} \end{cases}$$

(17)

where $k > 0$ is a parameter, the sensitivity to which is studied empirically in Sect. 4. $\text{maf}$ is 1 if the pixel is in or borders a constant region or if all motion residuals are large.

#### Complementary data term:

The cost for $x \in R_i$ is

$$f_i(x) = (1 - \text{maf}(x))\text{Res}_i(x) - \text{maf}(x)\log p_{i,x}(I_t(x)),$$

(18)

where $p_{i,x}$ are local normalized color histograms of the image $I_t$ within the region $R_i$. Therefore, if the motion is reliable, as defined by the maf, the cost is the residual of the pixel in the region and if the motion is unreliable, the cost is the fidelity of the pixel to the local intensity distribution of the region $R_i$. The data energy for region $R_i$ is then:

$$E_{\text{data}} = \int_{R_i} f_i(x) \, dx.$$ 

(19)

This complementary data term is a key feature in resolving disocclusions (Fig. 3).

### 3.2. Temporal and Spatial Regularity

To leverage temporal and spatial regularity of the regions, we first note that the warps are regular by construction within the Sobolev framework. We also note that, in between frames, disoccluded regions are small, adjacent to the object they belong to, and typically result in an updated region of similar shape. Thus, if $R_i^t$ is the forward warping of the $i$th region from frame $t$ to $t + 1$, we bias the final regions $R_i$ to be close to $R_i^t$ in shape and location.

To this end, we construct a local shape similarity prior. Measuring the similarity of $R_i$ and $R_i^t$ generally requires knowledge of point correspondences. Similar to ICP [5], we assume that $x \in R_i$ corresponds to its closest point in $R_i^t$, $c_{i,x}(x)$, which can be computed efficiently with Fast Marching [24]. Define the local shape similarity, $S_i : R_i \rightarrow \mathbb{R}^+$, of $R_i$ within the ball $B_{x,r}$ to $R_i^t$ within $B_{r,c_{i,x}(x)}$ as follows:

$$S_i(x) = \frac{1}{|B_{x,r}|} \int_{B_{x,r}} |1_{R_i}(y) - 1_{R_i^t}(c_{i,x}(x) - x + y)| \, dy,$$

(20)
In addition, to bias regions \( R_i \) towards being close to \( R'_i \), let \( d_{R'_i} \) denote the distance function to \( \partial R'_i \), and define

\[
E_{\text{dist}}^{i} = \int_{R_i} d_{R'_i}(x) \, dx.
\]

Finally, we induce spatial regularity of \( R_i \), \textit{i.e.}, nearby points \( x \) and \( y \) are penalized if they do not belong to the same region. Let

\[
W_{R_i} = G_s \ast (1 - 1_{R_i})
\]

be a Gaussian smoothing of standard deviation \( s \) of the complement of the indicator function of \( R_i \) \[11\]. A large value of \( W_{R_i}(x) \) implies that \( x \in R_i \) is near many points of \( D \setminus R_i \). We induce spatial regularity of \( R_i \) by

\[
E_{\text{smooth}}^{i} = \int_{R_i} W_{R_i}(x) \, dx.
\]

### 3.3. Overall Model and Optimization Method

The assumptions underlying our model are captured by the following energy, which is minimized with respect to the regions \( R_i \):

\[
E_{\text{seg}} = \sum_{i=1}^{N} E_{\text{data}}^{i} + \gamma_{a} E_{\text{shape}}^{i} + \gamma_{d} E_{\text{dist}}^{i} + \gamma_{s} E_{\text{smooth}}^{i},
\]

where \( \gamma_{a}, \gamma_{d}, \gamma_{s} > 0 \) are weights. We optimize the energy above by a first order approximation to the gradient descent, ignoring terms that involve integrals over \( R_i \). They could be easily included, at a high computational cost and modest performance gain. By defining

\[
H_i(x) = f_i(x) + \gamma_{a} S_i(x) + \gamma_{d} d_{R'_i}(x) + \gamma_{s} W_{R_i}(x),
\]

we arrive at our optimization scheme in Algorithm 2.

**Algorithm 2 Assigning Disocclusion to Regions**

1: // initialize \( R_i \) for gradient descent
2: Compute propagation of segmentation, \( R'_i \) using (27)
3: Compute disocclusion \( D = D \setminus \cup_i R'_i \)
4: Compute warps of \( R'_i \) using Algorithm 1
5: Compute \( H_i \) by substituting \( R_i \) with \( R'_i \cup D \)
6: Set \( R_i = R'_i \cup \{ x \in D : H_i(x) \leq H_j(x), \forall j \} \)
7: // end initialize
8: repeat // first order approximation of gradient descent
9: Update warps of \( R_i \) using Algorithm 1
10: Compute \( H_i \)
11: \( R_{\text{new}} = \{ x \in D : d_{R_i}(x) < \varepsilon, H_i(x) \leq H_j(x), \forall j \} \)
12: Update regions by \( R'_i = R_{\text{new}} \)
13: until \( R_i \)'s do not change between iterations

Algorithm 2 first computes an initialization of regions \( R_i \) to the gradient descent (lines 2-6). This is accomplished by
3.4. Initialization for the First Frame

So far we have assumed that, at time \( t \), we have a partition at time \( t - 1 \). This is the case during regime operation when processing a video sequence, but not when \( t = 0 \). For certain applications, such as interactive video segmentation [3, 2], one can assume that the user provides an initial partition. More in general, a number of methods could be employed to obtain an initial partition, using a variety of cues, including semantic labeling from trained detectors. While this process may be costly, it only needs to be performed once as our method affords us the ability to correct initial errors based on motion and appearance regularity.

In the next section, we present results for an initialization performed by clustering optical flow (with regularity (24) using Classic-NL [26]) during a longer initial temporal segment, until enough motion is observed (see Fig. 6).

4. Experiments

Our algorithm aims to segment objects, thus we test it on benchmarks with ground truth object annotation: the Freiburg-Berkeley Motion Seg. (FBMS-59) [23], and SegTrack (v1 & v2) [31, 20]. FBMS-59’s two sets - training (29 sequences) and test (30 sequences), range between 19-800 frames with multiple objects. SegTrack v2 consists of 14 sequences ranging from 29-279 frames with multiple objects. SegTrack v1 is an earlier version with single objects, which we use to expand the comparison to more methods.

Evaluation: FBMS-59 scores a subset of frames (3-41). Results are reported in terms of precision, recall, \( F \) measure, and the number of objects with \( F \geq 0.75 \). SegTrack (v1 & v2) evaluates, on all frames, the number of pixels incorrectly classified (v1). Results on v2 are reported as average intersection over union overlap.

Comparisons: On FBMS-59, we compare against a baseline approach [14], one based on clustering motion tracks [23], one segmenting based on occlusion, motion and appearance cues [1], and finally a most recent one integrating motion, appearance, occlusion, and temporal regularity [30]. On SegTrack, we compare to [8] that attempts to solve disocclusions using only appearance and to other state-of-the-art methods [20, 19, 21, 16, 34].

Initialization: On FBMS-59, we report results of our method automatically initialized as described in Sect. 3.4. On SegTrack our method is initialized by the user in frame 1 and compared with similarly initialized methods and also automated methods. Typically, sequences in SegTrack do not have enough object motion in the first few frames to ensure proper initialization.

Parameters: For FBMS-59, we tune the parameters on a few sequences in the training dataset, and then fix them on training and test datasets. On SegTrack, parameters are fixed. Parameters consistent across datasets are \( \gamma_{ls} = 0.1, \gamma_d = 0.001, \gamma_s = 5 \). Sensitivity of key parameters is addressed later.

Results on FBMS-59 are in Table 1. Figure 7 shows some representative outcomes. Overall our method is more accurate, even compared to non-causal (NC) methods that process the video in batch. This suggests that good disocclusion is key to accurate object segmentation.

Failure Cases on FBMS-59: The main source of error is the automatic initialization in frame 1. This could be mitigated by running our method on multiple candidate initializations, although initialization is not our focus here. To show that better initialization would resolve failures, we show that the results of the 10 most inaccurate cases (typically when an object failed to be detected) improves with user annotation in the first frame (Table 2, Figure 8). Fig. 9 shows that our method recovers from errors in the first frame (short of failed detection).

Figure 7. Sample Visual Results on FBMS-59. Comparison of various state-of-the-art methods. Only a single frame on various sequences are shown. Failure cases (bottom two) in our method typically arise when not enough motion is present in the first few frames.

<table>
<thead>
<tr>
<th>Training set (29 sequences)</th>
<th>Test set (30 sequences)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
</tr>
<tr>
<td>[14]</td>
<td>79.17</td>
</tr>
<tr>
<td>[23]</td>
<td>81.50</td>
</tr>
<tr>
<td>[1]</td>
<td>87.20</td>
</tr>
<tr>
<td>[30]</td>
<td>85.00</td>
</tr>
<tr>
<td>[30]-NC</td>
<td>83.00</td>
</tr>
<tr>
<td>ours</td>
<td>89.53</td>
</tr>
</tbody>
</table>

Table 1. FBMS-59 results. Average precision (P), recall (R), F-measure (F), and number of objects detected (N) over all sequences in the training and test datasets of FMS-59. Higher values indicate superior performance. All methods are fully automatic. [1], [30] and our method are causal; other methods are not.

<table>
<thead>
<tr>
<th></th>
<th>marple9</th>
<th>cats4</th>
<th>farm1</th>
<th>goats1</th>
<th>giraffes1</th>
<th>all</th>
</tr>
</thead>
<tbody>
<tr>
<td>ours (auto)</td>
<td>0.7950</td>
<td>0.7223</td>
<td>0.6730</td>
<td>0.6186</td>
<td>0.7515</td>
<td>0.7227</td>
</tr>
<tr>
<td>ours (manual)</td>
<td>0.9782</td>
<td>0.9025</td>
<td>0.7519</td>
<td>0.7505</td>
<td>0.9255</td>
<td>0.8617</td>
</tr>
</tbody>
</table>

Table 2. Failure cases on FBMS-59 in Fig. 7 can be enhanced with user annotation in the first frame. Thus, the main source of error in our method is the initialization. Results are in terms of F-measure.

Forward-Backward Sweeps on FBMS-59: Although disocclusions are backward-occlusions, addressed extensively in the literature [29, 1], computing disocclusions via forward-backward sweeps followed by a grouping procedure does not perform as well as our method. We compare to the non-causal version of [30], consisting of one forward and one backward pass. Then, advanced grouping is performed based on motion, appearance, temporal continuity, and constraints imposed by occlusions/disocclusions. The result, labeled [30]-NC in Table 1, is worse than ours on all measures. This reaffirms that forward-backward sweeps is not an adequate approach to resolve disocclusions.

Results on SegTrack: Table 3. We let the user annotate the first frame, as in [16, 8, 34]. Our method outperforms all others on all but one sequence. That our method outperforms [8] reaffirms that our exploiting complementary motion and appearance cues is beneficial. Results on v2 (Table 4, Fig. 10) show that our method out-performs fully automated ones but also those using user annotation.

Figure 8. Sample failure cases (various frames) on FBMS-59 in Fig. 7 are enhanced with user annotation in the first frame.
Recall
Precision
Precision vs. Recall Curve
Ball Size : 12 (pixels)
Threshold : 0.0012 to 0.0472

Recall
Precision
Threshold : 0.003
Ball Size : 3 to 30 (pixels)

Table 3. SegTrack v1 results. Evaluation is performed in terms of the number of pixels classified incorrectly; smaller values indicate superior results. Note that our method, [34], [16], and [8] use user annotation in frame 1, and [21], [19] do not.

Table 4. SegTrack v2. The evaluation is performed in terms of the overlap of the best segments; larger values indicate superior results. Our method and [34] uses user annotation in frame 1.

Sensitivity to Key Parameters: These include the ball size $r'$ and the threshold parameter $k$ in our textureless region detector (12) and (17). To this end, we plot PR curves (measured in terms of correct/incorrectly classified pixels) by fixing one parameter and varying the other and vice-versa. Results (Fig. 11) on the cereal box and statue sequences show that within the operating range, precision does not drop much as recall is increased.

Computational cost and implementation: Our unoptimized C++ implementation is available\(^2\). The costliest component is solving for the warps. This requires solving a linear PDE, for which there are many available fast-solvers that could be leveraged. We used conjugate gradient, which can be sped up. The overall cost of our algorithm varies with the amount of deformation between frames. Using a 3.1GHz 12-core processor (with parallelization for the gradient (7)), processing one frame on FBMS-59 takes on average 30 secs.

5. Discussion

We propose a method for handling disocclusion in object tracking that does not require explicit motion regularization, operates naturally in a coarse-to-fine framework, and leverages complementary motion and appearance cues. Our method exhibits reduced dependency on tuning parameters than competing ones, and mitigates typical failures modes.

Our approach assumes that a current estimate of the partition into objects is given at time $t$ to infer the same at $t+1$. If the given partition is nonsensical, most likely so will be the output of our inference scheme. This issue is particularly cogent at time $t = 0$. It can be addressed by spawning multiple trackers corresponding to different initialization hypotheses, later aggregating them through a voting scheme. In many tracking applications, however, the user decides what s/he wants to be tracked, so at least a rough initial partition is available. This is the case for interactive video post-processing [3]. In this case, it would be best to process the entire sequence non-causally, although in some cases processing a sliding batch is still desirable to avoid excessive delay in the interaction with the user.

Real-time operation remains a challenge, but our method has potential since we process data causally, and we use optimization methods that are rapidly evolving, so we can benefit from their improvements.

Acknowledgements

This work was partially supported by ARO W911NF-15-1-0564/66731-CS, AFOSR FA9550-15-1-0229, and KAUST OCRF-2014-CRG3-62140401.

\(^2\)Code: https://github.com/ycyang12/SODVS

Figure 10. Sample SegTrack v2 results of our method.
References


[34] L. Wen, D. Du, Z. Lei, S. Z. Li, and M.-H. Yang. JOTS: Joint online tracking and segmentation. In *CVPR*, 2015. 6, 7, 8


