Tracking indistinguishable translucent objects over time using weakly supervised structured learning

Luca Fiaschi[1], Ferran Diego[1], Konstantin Gregor[1], Martin Schiegg[1], Ullrich Koethe[1], Marta Zlatic[2] and Fred A. Hamprecht[1]

[1]HCI University of Heidelberg, Germany, http://hci.iwr.uni-heidelberg.de
[2]HHMI Janelia Farm, USA, http://janelia.org/

Abstract

We use weakly supervised structured learning to track and disambiguate the identity of multiple indistinguishable, translucent and deformable objects that can overlap for many frames. For this challenging problem, we propose a novel model which handles occlusions, complex motions and non-rigid deformations by jointly optimizing the flows of multiple latent intensities across frames. These flows are latent variables for which the user cannot directly provide labels. Instead, we leverage a structured learning formulation that uses weak user annotations to find the best hyper-parameters of this model.

The approach is evaluated on a challenging dataset for the tracking of multiple Drosophila larvae which we make publicly available. Our method tracks multiple larvae in spite of their poor distinguishability and minimizes the number of identity switches during prolonged mutual occlusion.

1. Introduction

In recent years, brilliant studies have emerged from the marriage of behavioral biology to sophisticated multiple-object tracking algorithms, e.g. [5, 14, 6]. The reported method is motivated by an investigation into the social dynamics of *Drosophila* larvae, a popular model organism in biology. These experiments require visual tracking from a single camera view of multiple interacting individuals moving on a well plate. While these controlled experimental conditions facilitate detection and segmentation of isolated individuals, the main challenge is the identification and tracking of the animals while they are touching or occluding one another over several frames. In this scenario, two aspects violate the assumptions of state of the art tracking algorithms that exploit object appearance features (e.g. [9, 30, 23, 11]) and/or motion models (e.g. [18, 7]) in order to handle mutual occlusions:

- Larvae are poorly distinguishable, translucent and mostly untextured objects of deformable shape.
- Larvae, when in contact, can exhibit an erratic motion.

Our formulation handles complex motion and non-rigid deformations without assuming the tracked objects to be distinguishable. At the core of our approach, we model the observed image intensities during mutual occlusion as a mixture of the latent intensities of each individual object. Thus, we disambiguate object identities by jointly optimizing the movement of multiple latent flows of intensity masses. This leads to a highly flexible model with many parameters that balance costs derived from generic low level cues. Rather than manually tuning these parameters, an important difference between our approach and [9, 30, 23, 11, 18, 7] is that we exploit structured learning with latent variables [28] to learn optimal weights from training annotations. Our main contributions are:

- The first formulation of the multiple object tracking problem which is targeted to disambiguate mutually occluding identical, deformable and translucent objects.
- A weakly supervised structured learning strategy to parametrize the corresponding energy terms that requires minimal user effort: only two clicks on each object before and after a training occlusion event.
- A solution to low density tracking of *Drosophila* larvae which minimizes the number of identity switches. We benchmark our approach on a challenging large dataset that we make publicly available for future studies[1].

[1]Download at http://hci.iwr.uni-heidelberg.de/Benchmarks

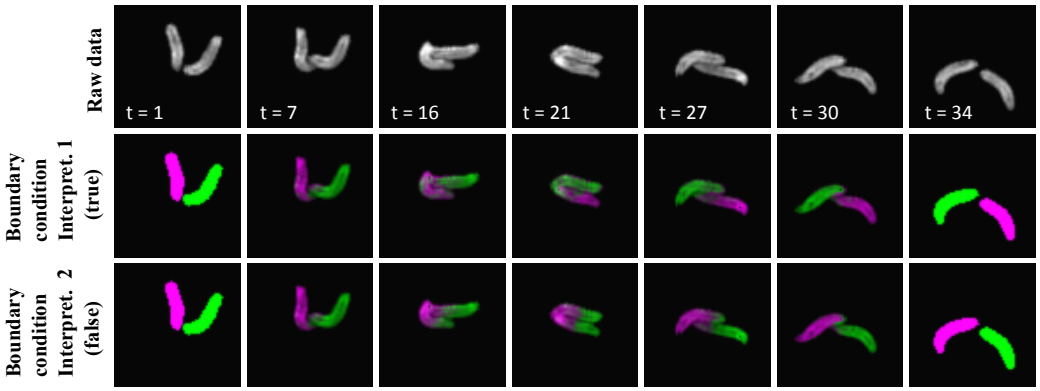

Figure 1. Top row: selected sub-frames from the raw data. Center and bottom rows: two possible interpretations of the sequence. The interpretations of the object identities are encoded in terms of boundary conditions (here shown in saturated colors, frame 1 and 34). Under our model (Eq. (1)), **inference** allows the estimation of the latent variable states (shown with desaturated colors) and gives the energy of the interpretation associated with the current boundary condition. Given training data, again in terms of boundary condition, **learning** finds model parameters so that the correct interpretation (central row) has lower energy than any erroneous interpretation.

2. Related Work

Multiple object tracking from a single camera view has a long history in computer vision where most of the work has focused on pedestrian and vehicle tracking. State of the art approaches that handle mutual occlusion of such targets often leverage two key characteristics of the tracked objects: (i) they are distinguishable, for example pedestrians with different clothes or cars of different colors, and/or (ii) they have easily predictable trajectories (such as are well described by linear motion models). These two characteristics have been exploited in frame-by-frame tracking by combined motion and appearance models like [18], and more recently by global data association techniques. These obtain state of the art performance by integrating over time local appearance and motion cues in order to jointly optimize the identity assignments to candidate detections, e.g. [11, 30, 7, 23]. Occluded and merged objects are commonly handled as misdetections. In [2, 9] the authors propose a global inference scheme in order to determine merged detections and then match the objects before and after occlusion using costs derived from appearance and motion features. Similarly to that approach, we build on our previous work [8] where we showed how larvae occlusions can be reliably detected. Differently from [9], we avoid relying on discriminative object features and instead seek a minimum action interpretation of the scene (expressed through flow of latent intensities) that is compatible with the observations.

Animal tracking and segmentation: The recent demand for automatic analysis of images in biology poses a new set of problems to the tracking community that are rarely found in natural images. In fact, objects of interest in biology are, on the one hand, highly deformable, often indistinguishable and sometimes translucent; and on the other hand they can exhibit motion that appears stochastic and they may divide into multiple parts. In biological image analysis, most of the research that explicitly handles overlapping and deformable objects has focused on frame-by-frame tracking of blob-like structures such as cells [15, 3, 22]. In [15], merged cells are separated by combining level sets with a motion filter, while [3] uses frame-by-frame partial contour matching and [22] a Gaussian mixture model. Excellent work exists on mice tracking [5] using particle filters to keep track of the object contours but this approach is limited to represent affine transformations of a small number of manually designed shape templates. Particle filters have also been used for ants tracking [14], but simpler and independent constant velocity models are implemented in most software packages, such as `Ctrax` [6] for Drosophila. A recent review of open-source worm like object trackers is [10]. According to the authors, all these softwares work in uncrowded situations and do not handle occlusion. Tracks are simply terminated on mutual occlusion events and reinitialized afterwards while identity errors are often manually edited [6]. Wählby [26] has proposed a method for segmentation of overlapping worms. However, this approach relies on candidate segments produced by skeletonization and performs best for elongated objects which have a low probability to lie side by side.

Structured learning and transportation theory: The methods in [18, 5, 11, 30, 15, 3, 9, 23, 7] require parameters which are set manually. Our approach builds on the intuitions of Lou *et al*. that in [17] proposed to use weakly supervised structured learning [28] for parametrizing the energy of a data association model for cell tracking [16]. Following [17], we show how to learn our energy parameters

from weak user annotations. However, our application and energy formulation differ from [16, 17]. In these works objects identities were lost at occluded regions as only frame-by-frame assignments were considered.

Our model also draws ideas from the literature on Earth Mover Distance (EMD) [21]. EMD finds the minimum cost flow between two distributions of masses one of which is seen as the source and the other as the sink. This distance has several applications in tracking, e.g. [20, 19, 27]. Ren *et al*. in [20] introduced EMD for single object tracking and used it to model the observed flow of intensity. Oron *et al*. [19] showed that EMD costs can be updated online and Wang *et al*. [27] proposed to learn their values from training data. We extend these ideas by allowing the masses to take different colors, and we jointly minimize the cost of multiple flows in a fashion similar, but not identical to, a multi-commodity flow problem [1]. These are used in operations research to minimize the total shipment cost of multiple distinguishable products over the same network.

3. Problem Formulation and Modeling

Our key idea is sketched in Fig. 1. When two indistinguishable objects overlap and then separate again, we have two possible interpretations of their identity assignment. To build intuition, pretend that each object has a unique color but that we have only a monochrome sensor. In such a situation, the color of each pixel is a *latent* variable, while the pixel grey value intensities are *observed* data. Given a boundary condition determined by the current interpretation (i.e. the identity assigned to each individual before and after the occlusion), our model is an energy function which allows estimating the state of all latent variables as the minimum energy solution. The aim of learning is to find model parameters such that the correct interpretation obtains the lowest energy.

Our model is a generalization of EMD in the sense that the motion of multiple larvae is represented by the flow of multiple *differently* colored masses. These flows are jointly optimized across several frames, subject to costs that express the following notions:

- To obtain the most parsimonious interpretation of the action, colored masses should move around as little as possible, while still satisfying the boundary condition.
- The spatial distribution of each color should be smooth inside each object.
- The intensity of all colors in a pixel should sum up to approximate the observed overall intensity in that pixel.
- The conservation of colors over time holds only approximately, to account for overall fluctuations in the image intensity, as well as Poisson and sensor noise or sensor saturation.

In formulating this energy function, we make two design choices. First, by using a weighted sum of costs derived from multiple features, we make sure that the model is sufficiently expressive to allow assigning the lowest energy to the correct solution for each training sample. Second, we restrict our formulation to terms that are convex in the latent variables, thus allowing for efficient optimization.

3.1. Precise formulation

The energy function in Eq. (1) gives expression to the notions from the previous section (cf. Table 1). It depends on two kinds of variables: the *flows* $\{f\}$ and the *masses* $\{m\}$. If we index each pixel inside the spatiotemporal volume of the video as $i = 1, ..., N_P$, a colored mass variable m_i^k represents the intensity mass associated with larva $k = 1, ..., N_L$ at pixel i. Masses can be observable variables or latent variables of the problem. In particular, the masses of pixels inside an *isolated* object are observable variables because during the learning we may ask the user to annotate the entire isolated objects with a single stroke. Conversely, pixels of overlapping objects are associated with *latent* mass variables. In cases where we need to distinguish between observable and latent masses, we use respectively symbols $\dot{m}_i^k$ and $\mathring{m}_i^k$: $\{\dot{m}\} \cap \{\mathring{m}\} = \emptyset$ and $\{\dot{m}\} \cup \{\mathring{m}\} = \{m\}$. We reserve the symbol $m_i^\dagger$ to represent the measured grey value intensity of a pixel i and we collect the elements of the countable sets $\{\mathring{m}\}$, $\{\dot{m}\}$, $\{f\}$ in the vectors $\mathring{\mathbf{m}}$, $\dot{\mathbf{m}}$, $\mathbf{f}$.

$$
\begin{aligned}
E(\underbrace{\mathbf{f}, \mathring{\mathbf{m}}}_{\text{latent variables}} \;;\; \dot{\mathbf{m}}, \mathbf{w}) \;=\; & \underbrace{\sum_{i=1}^{N_P} \left| \sum_{k=1}^{N_L} m_i^k - m_i^\dagger \right| d_{ii}^1 \; w^1}_{\text{data fidelity term}} \\
& + \underbrace{\sum_{i=1}^{N_P} \sum_{k=1}^{N_L} \left| \sum_{j \in \mathcal{N}_i^{st}} f_{ij}^k - m_i^k \right| \; w^1}_{\text{outgoing flow conservation term}} \\
& + \underbrace{\sum_{j=1}^{N_P} \sum_{k=1}^{N_L} \left| \sum_{i \in \mathcal{N}_j^{st}} f_{ij}^k - m_j^k \right| \; w^1}_{\text{incoming flow conservation term}} \\
& + \underbrace{\sum_{k=1}^{N_L} \sum_{ij \in \mathcal{N}^s} |m_i^k - m_j^k| d_{ij}^2 \; w^2}_{\text{smoothness term}} \\
& + \underbrace{\sum_{l=3}^{M} \sum_{k=1}^{N_L} \sum_{ij \in \mathcal{N}^{st}} f_{ij}^k d_{ij}^l \; w^l}_{\text{flow cost term}}
\end{aligned}
\tag{1}
$$

To model motion, we define flows on a directed graph connecting pixels in the video. The graph connectivity is built on a spatiotemporal neighborhood $\mathcal{N}^{st}$ connecting pixels in consecutive frames. The flow variables f_{ij}^k, assigned to each edge $ij \in \mathcal{N}^{st}$, represent the flow of the mass associated with larva k between pixel i at time t and pixel j at time $t+1$. Flows are always latent variables of the problem and maintain temporal coherence between the pixels in consecutive frames.

The energy is a sum of terms weighted by w^l, $l = 1, ..., M$. These terms fall into four categories: from the bottom to the top of Eq. (1), *flow cost terms* penalize flow of mass between pixels which are different according to the parameter d_{ij}^l dependent on the l^{th} image feature. The costs d_{ij}^l can take into account not only spatial distance, as expressed by various powers of the Euclidean distance, but also dissimilarities in local appearance, *etc*. (see section 3.4). The *smoothness term* favors spatially smooth solutions where adjacent pixels have similarly colored masses. For each frame, this term is defined on pairs of pixels within the spatial neighborhood $\mathcal{N}^s$. The *data fidelity term* favors solutions where the sum of the colors associated with each pixel is close to the grey value intensity $m_i^\dagger$. Finally, the *flow conservation terms* enforce temporal consistency of proximate pixels at neighboring time steps. In classical flow problems these terms are imposed as linear constraints while we include these terms in the objective to account for fluctuations in the intensity. Although not required by the formulation, we reduce the number of parameters by giving the same weight w^1 to data fidelity and flow conservation terms.

Under the boundary condition given by observed mass variables $\dot{\mathbf{m}}$, the optimization of Eq. (1) allows estimating the latent masses and the latent flows, $\mathring{\mathbf{m}}$ and $\mathbf{f}$. The energy optimization problem

$$\underset{\mathbf{f}, \mathring{\mathbf{m}} \in \mathbb{R}_+^M}{\operatorname{argmin}} \; E(\mathbf{f}, \mathring{\mathbf{m}} \; ; \; \dot{\mathbf{m}}, \mathbf{w}) \tag{2}$$

is linear in the weights $\mathbf{w}$ and linearizable in the flows $\mathbf{f}$ and the masses $\mathbf{m}$ when replacing the absolute values with auxiliary variables. This is a standard technique, see [4]. Problem 2 is therefore a convex problem that can be solved efficiently by linear programming[2].

3.2. Inference: identity interpretation

Problem 2 requires the masses $\dot{\mathbf{m}}$ associated with the pixels of all isolated (non-overlapping) individuals as boundary conditions. In the training set, these boundary conditions are observed; at prediction time, they are merely observable (but not observed). To identify and track all objects, we seek to identify those boundary conditions that

[2]The optimization of Eq. (1) is implemented using ILOG Cplex.

Symbol	Definition
t	Time index $t = \{1, ..., T\}$
i, j	Pixel indices. There are N_p pixels inside the spatiotemporal volume of the video: $i, j \in \{1, ..., N_p\}$
$\mathcal{N}^s$	spatial neighborhood relates pixels in the same frame
$\mathcal{N}^{st}$	spatiotemporal neighborhood relates pixels in consecutive frames
l	Feature index, $l \in \{1, ..., M\}$
k	Mass color index, $k \in \{1, ..., N_L\}$
f_{ij}^k	Flow associated with larva/color k between pixels $i, j \in \mathcal{N}^{st}$
d_{ij}^l	Cost of flow between pixels $i, j \in \mathcal{N}^{st}$ computed from the l^{th} feature.
w^l	Weight associated with the l^{th} feature
m_i^k	Mass associated with larva/color k in pixel i. Masses can be: $\dot{m}_i^k$ observable variables $\mathring{m}_i^k$ latent variables
$m_i^\dagger$	Measured grey value intensity in pixel i

Table 1. Notation.

are most plausible according to our energy function. Valid boundary conditions must respect two constraints: first, each isolated object has all its pixels labeled entirely by masses of the same color; and second, each color is assigned to only one isolated object per frame. Methods such as [2, 8] break the entire video into non-occluded parts (easy to track) and occlusion events (difficult to track). At occlusion events, these two constraints ensure that each single object entering the occluded spatiotemporal region is matched to another single object leaving that region. If we define $\mathcal{I}$ as the space of such valid interpretations, then in order to find the lowest energy identity assignment of all objects, we solve the following optimization problem:

$$\underset{\substack{\dot{\mathbf{m}} \in \mathcal{I} \\ \mathbf{f}, \mathring{\mathbf{m}} \in \mathbb{R}_+^M}}{\operatorname{argmin}} \; E(\mathbf{f}, \mathring{\mathbf{m}}, \dot{\mathbf{m}} \; ; \; \mathbf{w}) \tag{3}$$

Rather than incorporating the constraints on $\mathcal{I}$ explicitly, we exploit the structure of the problem: we simply solve repeatedly for each of the $N_L^c!$ distinct interpretations that are possible for the identity assignment of those N_L^c objects that aggregate into a cluster. Note that, in most cases, N_L^c is only two or three in our data.

In summary, then, while Eq. (2) is convex, the additional constraints in Eq. (3) result in a non-convex optimization problem with a combinatorial number of local optima. We explore all of these for mutual occlusions of up to three objects. For four or more occluded objects, we resort to the approximation detailed in section 4.2.

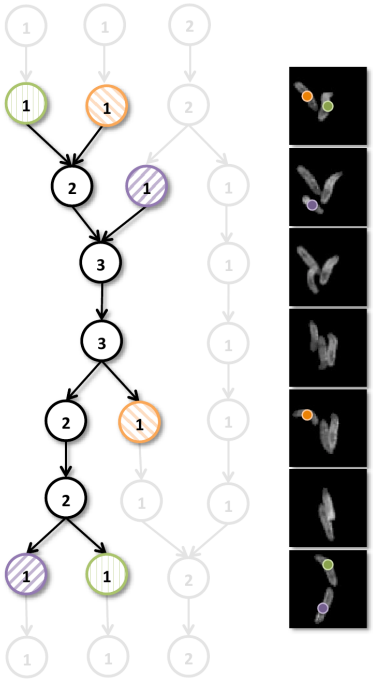

Figure 2. Illustration of a training example corresponding to an occluded spatiotemporal region. On the left side, the counting graph as obtained by the algorithm of [8]: each foreground connected component is depicted with a circle labeled with the number of contained objects. Bold lines indicate a subgraph where an occlusion of 3 larvae is detected. On the right side, the spatiotemporal region of interest corresponding to this subgraph is depicted. Note that in each frame, *only* the objects corresponding to subgraph's nodes are visible. On the left and right sides: the colors identify individuals which enter or leave occlusion as labeled by the human expert during the training phase. The human expert visualizes the sequence at higher temporal resolution and can therefore provide confident annotations.

3.3. Parameter learning with partial annotations

The learning determines the optimal weights $\mathbf{w}$ in Eq. (3) so that the correct interpretation of the training data receives lower energy than any wrong interpretation. We solve this energy parametrization problem via structural risk minimization [25, 28, 17]. Our training examples, $n = 1, ..., N$ for the observable colored mass variables are given in term of user annotations that mark the identity of isolated larvae only before and after the ambiguous region where they overlap, as depicted in Fig. 2. Arguably, this annotation requires a minimal click effort for the user.

A training example $(\dot{\mathbf{m}}_n, \mathring{\mathbf{m}}_n, \mathbf{f}_n, \mathbf{d}_n)$ is composed of observed and latent variables and by the features of the image region $\mathbf{d}_n$. The loss function of the learning problem penalizes positive differences between the energy of the correct interpretation and the lowest energy among any of the wrong interpretations. If we define $\bar{\mathcal{I}}_n = \mathcal{I}_n \backslash \dot{\mathbf{m}}_n$ as the set of valid but wrong boundary conditions then, similarly to [28], the loss function can be written as:

$$\begin{aligned} L(\mathbf{w}, \dot{\mathbf{m}}_n) \;&=\; \min_{\mathbf{f}, \mathring{\mathbf{m}} \in \mathbb{R}_+^M} E(\mathbf{f}, \mathring{\mathbf{m}}\,;\, \dot{\mathbf{m}}_n, \mathbf{w}) \\ &\quad - \min_{\substack{\dot{\mathbf{m}} \in \bar{\mathcal{I}}_n \\ \mathbf{f}, \mathring{\mathbf{m}} \in \mathbb{R}_+^M}} \Big[\, E(\mathbf{f}, \mathring{\mathbf{m}}, \dot{\mathbf{m}}\,;\, \mathbf{w}) - \Delta(\dot{\mathbf{m}}_n, \dot{\mathbf{m}}) \Big] \end{aligned} \tag{4}$$

Here, $\Delta(\dot{\mathbf{m}}_n, \dot{\mathbf{m}})$ is the task loss function which depends only on the observable variables. In this study, we use the number of identity switches between the predicted interpretation and the gold standard from the training set. Note that our restriction to the set of valid but wrong interpretations $\bar{\mathcal{I}}_n$ (see section 3.2) makes the optimization problem one that is not additive over local cliques of variables. The model is trained by finding the optimal weights which minimize the regularized structural risk:

$$\begin{aligned} &\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w}||_2^2 \;+\; \frac{C}{N} \sum_n |L(\mathbf{w}, \dot{\mathbf{m}}_n)|_+ \\ &\text{s.t.} \quad w^1, ..., w^M \geq 0 \end{aligned} \tag{5}$$

Here, $|\cdot|_+ := \max(0, \cdot)$ is the hinge function which ensures a positive contribution from the loss while additional constraints guarantee the convexity of Eq. (1).

The loss in Eq. (4) is non-convex but is instead the difference of two convex functions (DC) [24]. Following [28], we find a local minima to problem Eq. (5) by using the CCCP procedure [29]. Briefly, this iterative scheme alternates between estimating the most probable state for the latent variables and solving the structured SVM problem treating all variables as observed. We implement CCCP based on the n-slack [12] formulation of structured SVM[3]. Taken together, this allows us to learn the parameters of the model from very weak annotations and generic features.

3.4. Features

Multiple expressive features d_{ij}^l, ensure that the model in Eq. (1) can differentiate between the possible interpretations of the sequence. This is in contrast to previous work [9, 30, 23, 11, 7] whose cost functions were parametrized manually. All features are summarized in table 2. Beside traditional features such as the powers of the Euclidean distance computed from the spatial locations of two pixels, we propose a new set of features that captures the local spatiotemporal structure of the data. In particular, these features are derived from the intensity profile $\Phi_{ij}(s)$, $s \in [0, 1]$. This is computed for pixel i located in frame t and pixel j located in frame $t + 1$, along segment parametrized by s connecting the spatial locations of the two pixels. As

[3] Our implementation builds on the open-source library Pystruct.

in table 2, we collect the features in three groups: the first group are the features multiplying the smoothness and the data fidelity term, the second group are all the purely spatial costs and the third group includes the proposed features derived from $\Phi_{ij}(s)$. The learned weights of these features are shown in Fig. 3.

Feature	
d^1_{ii}	$\exp\left(-m_i^\dagger/255\right)$
d^2_{ij}	constant
d^3_{ij}	Spatial Euclidean distance between pixels i, j
d^4_{ij}	Second power of d^3_{ij}
d^5_{ij}	Fourth power of d^3_{ij}
d^6_{ij}	$\lvert m_i^\dagger - m_j^\dagger \rvert$
d^7_{ij}	$\int_0^1 \Phi_{ij}(s) ds$
d^8_{ij}	$\max_{s\in[0,1]} \Phi_{ij}(s) - \min_{s\in[0,1]} \Phi_{ij}(s)$
d^9_{ij}	$\langle \left(\Phi_{ij}(s) - \langle \Phi_{ij}(s) \rangle\right)^2 \rangle$
d^{10}_{ij}	$\int_0^1 I_{\{\Phi_{ij}(s) \leq \text{background}\}} ds$

Table 2. Definition of used features. I is an indicator function and the threshold for the background intensity is set to 50.

4. Experiments

In our experiments, we compare the proposed method to the related work [6, 16, 9, 22]. Firstly, we use the established software `Ctrax` [6] that can track multiple *Drosophila* adults. `Ctrax` uses frame-by-frame tracking based on a constant velocity model and also incorporates automatic corrections for merged detections by considering multiple splitting hypotheses. Secondly, we combine ideas from [9] with [16] in a method that we term L-BM, in order to enhance bipartite matching used by [9]. Briefly, as in [9] we compute the minimum cost bipartite matching between the isolated larvae entering the encounter region and those exiting the encounter region. In addition, costs are computed as linear combination of three features: Euclidean distance between the centers of the isolated larvae, difference in size and difference in average intensity; as in [16] optimal weights are learned with a structured SVM. Thirdly, we compare with Conservation Tracking (CT) [22]. Like our proposal, CT initially detects clusters of occluding objects and then disambiguate the occlusions. In contrast to our work, CT fits a Gaussian mixture model for each time to each occluded region and then performs tracking by data association. For a fair comparison with our method, we initialize both algorithms with the results from [8] and optimize the parameters of [22] via grid search.

4.1. Dataset and evaluation metric

We have evaluated our approach on a challenging dataset of larvae tracking composed of 33 high resolution movies.

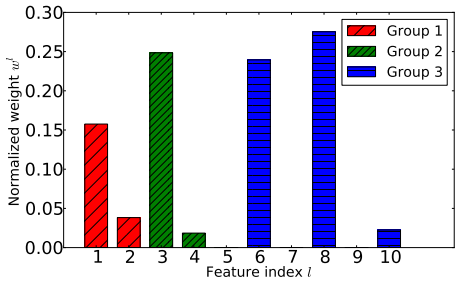

Figure 3. Parameters $\mathbf{w}$ learned from the training data and normalized to sum to one.

Each movie has a length of 5 minutes, a temporal resolution of 3.3 frames per second (1000 frames in total) and contains on average 20 larvae. The spatial resolution is 135.3 μm/pixel at 1400×1400 pixels/image. For the preprocessing and the construction of the counting graph we follow the guidelines from [8]. That counting algorithm obtained a precision of 99.9% on this dataset, indicating an almost perfect tracking for the isolated larvae. We therefore focus our evaluation on occlusion events that result from animals overlapping each other or from undersegmentation. We extract all subgraphs containing clusters of two or more larvae from the counting graph. These regions are sparse and correspond mostly to the occlusion of two larvae. A human expert manually labeled each larva entering or leaving each region in order to create a gold standard, as sketched in Fig. 2. Similarly to [13], performance is measured by counting the number of identity mismatches between the output of the algorithm and the gold standard. Missed detections and lost tracks are added to the error. This metric is then normalized by the total number of objects entering the region: 1478 for regions with two larvae, 96 for three larvae and 28 for four or more larvae.

4.2. Implementation and experimental details

Three approximations were made in order to reduce the computational effort of the proposed method. First, for each encounter we select up to 15 sub-frames linearly spaced in time in order to reduce the number of variables involved in the optimization. Second, the threshold for the spatiotemporal neighborhood $\mathcal{N}^{st}$ is adaptively chosen for each encounter as the minimum distance such that each foreground pixel is connected by at least one temporal edge (10 pixels on average). The spatial neighborhood $\mathcal{N}^s$ is fixed to a radius of 1 pixel. Third, when four or more larvae are in the overlapping region, L-BM is used to retrieve only the first six interpretations with lowest matching cost. Among these, with our main method, we select the interpretation which obtains the lowest energy.

Our main proposal as well as L-BM are trained using only 25 examples of encounters of two larvae (3% of all

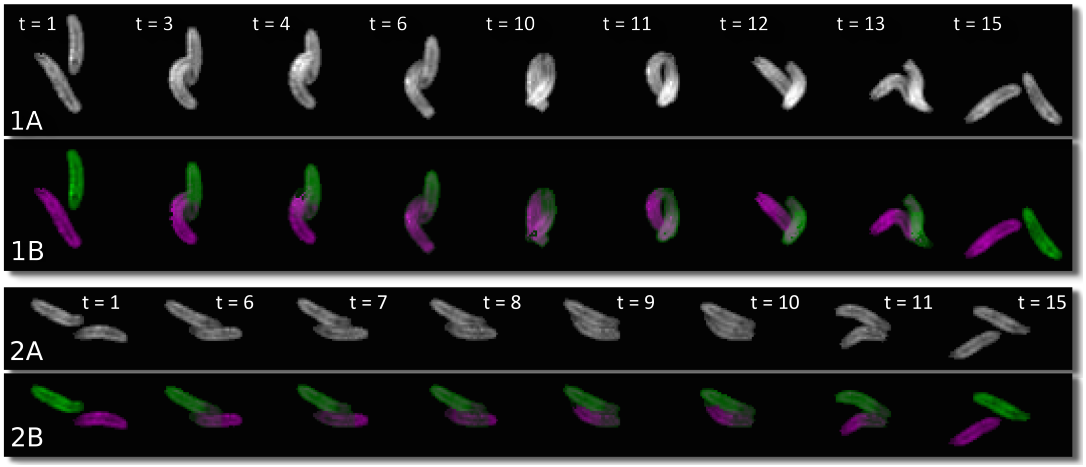

Figure 4. Two examples from the test set. Raw data (1A, 2A) and lowest energy interpretation inferred by our algorithm (1B, 2B). In frames 1 and 15 the objects are separated and the value for the masses is set by the boundary conditions. In intermediate frames, the sum of the two colors is close to the grayvalue intensity in the raw data and influenced from the smoothness term of Eq. (1).

Method	Total	$N_L^c = 2$	$N_L^c = 3$	$N_L^c \geq 4$
Random Guess	51.5%	50.0%	66.7%	76.1%
Ctrax [6]	13.2%	11.5%	26.3%	57.1%
L-BM	11.8%	10.5%	23.1%	42.8%
CT [22]	9.4%	6.7%	34.7%	64.3%
This proposal	**5.3%**	**4.2%**	**14.7%**	**32.1%**

Table 3. Identity assignment error across the dataset: breakdown into encounters of two, three, four and more larvae, and average weigthed by the number of encounters per type.

available encounters) and testing is performed on the entire dataset. Most encounters have a very short duration, so to harvest more hard examples, we perform a first inference round and then train the model again with a mixture of 15 randomly selected and 10 hard examples. To assess the variability in performance with a randomly selected training set, the learning curve of the algorithm with different number of training examples is included in the supplementary material. Training our model takes around 8 hrs while the median inference time is 1.3hr/movie on a 2.4GHz Intel Quad-Xeon machine. 90% of the time is spent on encounters of 3 or more larvae. Further details on the running time are relegated to the supplementary.

4.3. Results and discussion

Fig. 4 and Fig. 5 qualitatively illustrate our results. For two challenging cases from the test set, Fig. 4 shows the inferred state of the latent masses of the interpretation with lowest energy. Fig. 5 compares the inferred states of the latent variables for the six possible interpretations of an encounter of three larvae. Due space limitations, further results shown at higher temporal resolution as well as the reconstructed tracking of entire movies are shown in the supplementary material.

The quantitative comparison between our approach and other methods is presented in table 3. Both L-BM and `Ctrax` have similar performance on this dataset. However L-BM is less prone to misdetections and learns its parameters from training examples. For encounters of two or three larvae, our main proposal produces consistently more accurate results without using any appearance feature. On encounters of four or more larvae, where L-BM is used to retrieve candidate interpretations, our approach can improve the results without however reaching truly satisfactory performance. For all methods, performance decreases with the number of larvae in the encounter as the scene becomes more cluttered and the number of interpretations increases. In particular, [22] which assumes roughly Gaussian shapes, offers a computationally less expensive alternative to our method for occlusions of two larvae but it has difficulties disambiguating clusters of more than two entangled articulated targets, both qualitatively and as indicated by the numbers in table 3.

On this dataset, the accuracy of our approach outperforms the compared methods. Even so, for some applications the resulting error for agglomerates of more individuals is still high, and in those settings computation time also needs to be improved further.

In summary, we have presented and evaluated a new tracking algorithm that specifically addresses the identity switching problem for the challenging situation of multiple indistinguishable translucent overlapping objects. These are allowed to be deformable and perform complex motions, such as crawling across each other. All in all, Eq. (1) expresses a "least action principle" – after appro-

priate parametrization, a scene can be interpreted in terms of a minimal-cost transformation, or transport of masses of different colors. The structured learning framework allows to fine-tune the costs for these transportation processes so as to make the true solutions, as given by the training set, the least costly. A training set can be compiled very conveniently by specifying the identity of individual larvae only before they enter or after they leave an agglomerate. We solve the associated hard latent variable learning problem, and achieve encouraging results on challenging sequences of social larvae. While the computational complexity of our approach does not yet scale to large populations, it opens new avenues for the study of social interactions in animals.

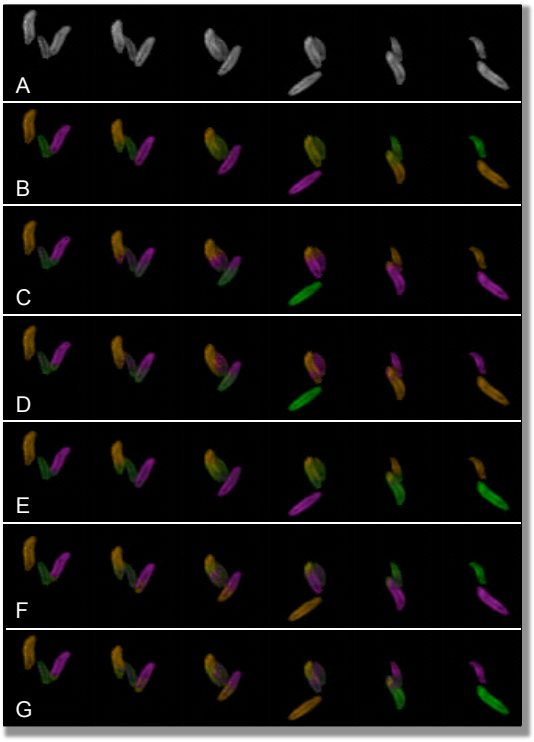

Figure 5. Selected sub-frames from an encounter of 3 larvae. From the top: raw data (A) and the 6 possible interpretations ranked by increasing energy (from B to G). Our approach correctly assigns the lowest energy to the interpretation in the second row (B).

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice Hall, 1993. 3

[2] B. Bennett, D. R. Magee, A. G. Cohn, and D. C. Hogg. Using spatio-temporal continuity constraints to enhance visual tracking of moving objects. In *ECAI*, 2004. 2, 4

[3] R. Bise, K. Li, S. Eom, and T. Kanade. Reliably tracking partially overlapping neural stem cells in dic microscopy image sequences. In *MICCAI-OPTIMHisE*, 2009. 2

[4] J. Bisschop. *AIMMS-Optimization modeling*. Paragon Decision Technology, 2006. 4

[5] K. Branson and S. Belongie. Tracking multiple mouse contours without too many samples. In *CVPR*, 2005. 1, 2

[6] K. Branson, A. A. Robie, J. Bender, P. Perona, and M. H. Dickinson. High-throughput ethomics in large groups of drosophila. *Nature methods*, 2009. 1, 2, 6, 7

[7] R. T. Collins. Multitarget data association with higher-order motion models. In *CVPR*, 2012. 1, 2, 5

[8] L. Fiaschi, G. Konstantin, B. Afonso, M. Zlatic, and F. A. Hamprecht. Keeping count: leveraging temporal context to count heavily overlapping objects. In *ISBI*, 2013. 2, 4, 5, 6

[9] J. Henriques, R. Caseiro, and J. Batista. Globally optimal solution to multi-object tracking with merged measurements. In *ICCV*, 2011. 1, 2, 5, 6

[10] S. J. Husson, W. S. Costa, C. Schmitt, A. Gottschalk, et al. Keeping track of worm trackers. *WormBook: the online review of C. elegans biology*, 2012. 2

[11] H. Jiang, S. Fels, and J. J. Little. A linear programming approach for multiple object tracking. In *CVPR*, 2007. 1, 2, 5

[12] T. Joachims, T. Finley, and C.-N. J. Yu. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009. 5

[13] B. Keni and S. Rainer. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008. 6

[14] Z. Khan, T. Balch, and F. Dellaert. MCMC data association and sparse factorization updating for real time multitarget tracking with merged and multiple measurements. *TPAMI*, 2006. 1, 2

[15] K. Li, M. Chen, T. Kanade, E. D. Miller, L. E. Weiss, and P. G. Campbell. Cell population tracking and lineage construction with spatiotemporal context. *Medical image analysis*, 12(5):546, 2008. 2

[16] X. Lou and F. A. Hamprecht. Structured learning for cell tracking. In *NIPS*, 2011. 2, 3, 6

[17] X. Lou and F. A. Hamprecht. Structured learning from partial annotations. In *ICML*, 2012. 2, 3, 5

[18] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: multitarget detection and tracking. In *ECCV*, 2004. 1, 2

[19] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan. Locally orderless tracking. In *CVPR*, 2012. 3

[20] X. Ren and J. Malik. Tracking as repeated figure/ground segmentation. In *CVPR*, 2007. 3

[21] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. In *ICCV*, 2000. 3

[22] M. Schiegg, P. Hanslovsky, B. X. Kausler, L. Hufnagel, and F. A. Hamprecht. Conservation Tracking. In *ICCV*, 2013. 2, 6, 7

[23] B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Multi-commodity network flow for tracking multiple people. *TPAMI*, 2013. 1, 2, 5

[24] P. D. Tao and L. T. H. An. Convex analysis approach to DC programming: theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997. 5

[25] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(2):1453, 2006. 5

[26] C. Wählby, T. Riklin-Raviv, V. Ljosa, A. L. Conery, P. Golland, F. M. Ausubel, and A. E. Carpenter. Resolving clustered worms via probabilistic shape models. In *ISBI*, 2010. 2

[27] F. Wang and L. J. Guibas. Supervised earth mover's distance learning and its computer vision applications. In *CVPR*, 2012. 3

[28] C.-N. J. Yu and T. Joachims. Learning structural SVMs with latent variables. In *ICML*, 2009. 1, 2, 5

[29] A. L. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15(4):915–936, 2003. 5

[30] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008. 1, 2, 5