Computer Vision and Image Understanding 000 (2015) 1–13

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

# Weakly supervised motion segmentation with particle matching☆

Hodjat Rahmati[a,*], Ralf Dragon[b], Ole Morten Aamo[a], Lars Adde[c,d], Øyvind Stavdahl[a], Luc Van Gool[b]

[a] Department of Engineering Cybernetics, NTNU, Trondheim, Norway
[b] Computer Vision Lab, ETH, Zurich, Switzerland
[c] Clinic for Clinical Services, St. Olavs University Hospital, Trondheim, Norway
[d] Department of Laboratory Medicine, Children and Woman's Health, Faculty of Medicine, NTNU, Trondheim, Norway

ARTICLE INFO

*Article history:*
Received 24 February 2015
Accepted 12 July 2015
Available online xxx

*Keywords:*
Motion segmentation
Particle matching
Tracking
Cerebral palsy
Computerized diagnosis

ABSTRACT

Motion segmentation refers to the task of segmenting moving objects subject to their motion in order to distinguish and track them in a video. This is a challenging task in situations where different objects share similar movement patterns, or in cases where one object is occluded by others in part of the scene. In such cases, unsupervised motion segmentation fails and additional information is needed to boost the performance. Based on a formulation of the clustering task as an optimization problem using a multi-labeled Markov Random Field, we develop a semi-supervised motion segmentation algorithm by setting up a framework for incorporating prior knowledge into the segmentation algorithm. Prior knowledge is given in the form of manually labelling trajectories that belong to the various objects in one or more frames of the video. Clearly, one wishes to limit the amount of manual labelling in order for the algorithm to be as autonomous as possible. Towards that end, we propose a particle matching procedure that extends the prior knowledge by automatically matching particles in frames over which fast motion or occlusion occur. The performance of the proposed method is studied through a variety of experiments on videos involving fast and complicated motion, occlusion and re-appearance, and low quality film. The qualitative and quantitative results confirm reliable performance on the types of applications our method is designed for.



## 1. Introduction

In a still scene where objects are visually blended to the background, such as animal camouflage, it is difficult or impossible to distinguish the objects from the background. In such a situation, motion information is a strong clue for visual perception of the surrounding environment. Although the process of motion perception appears straightforward to the human visual system, it is a difficult problem from a computational perspective. Humans are continuously detecting, tracking and registering surrounding objects, and to them occlusion, disocclusion and different motion patterns seem less of an issue. Dissimilar to humans, these could be extremely challenging for a computer-based method.

An application where we have to deal with fast and complicated motion patterns is the analysis of an infant's motion. Such an analysis is needed frequently in the medical world, for example in our application that is to predict cerebral palsy (CP) at an early stage. To do so, we need to extract the motion information out of videos of infants at 2–4 month postterm age. Over the past few years, a number of computer-based movement assessment tools have been developed [1–3]. However, there are problems limiting their practical use. First, they must be installed in a controlled environment. Second, they use instrumentation that might affect the infant's body movements. Finally, experts are needed for interpretation and analysis of the results. Recently, our research group has been studying the prediction of CP using a normal 2D monocular camera by a simple frame differencing approach without any need for instrumentation on the infant [4–7]. While these early studies show great promises, the algorithms used are sensitive to lighting conditions, clothing and skin color. In addition, the video data is aggregated into movement variables or features that provide limited clinical insight [8].

To overcome these problems, we are interested in extracting motion information out of a video and analysing this motion information to separate moving objects (in our case the infant's body parts) from the background and from each other. In situations like our application, where the objects share similar motions, the motions are fast and with complicated patterns, and occlusion happens frequently, motion cannot be informative enough by itself. Thus, additional information is needed, in our case some prior knowledge about the

☆ This paper has been recommended for acceptance by Leonid Sigal.

* Corresponding author. Tel.: +47 450 88 591.
*E-mail address:* hodjat.rahmati@itk.ntnu.no (H. Rahmati).

http://dx.doi.org/10.1016/j.cviu.2015.07.004
1077-3142/

assignment of the trajectories. The prior knowledge in the present application is manual labeling of the infant's body parts in one or more frames of the video. Clearly, one wishes to minimize this manual labor such that the segmentation procedure is as autonomous as possible. Incorporating prior knowledge is challenging, and is addressed in Sections 3 and 4. In [9] we proposed an energy minimization technique to incorporate prior knowledge into the segmentation procedure. In the current paper, we improved this motion segmentation method by incorporating an multi-scale particle matching procedure into the method. As a result, less prior knowledge is needed.

## 2. Related work

Motion segmentation is the task of segmenting moving objects from an image sequence subject to their motion over time. While the term *segmentation* does not specify which information is used to obtain the result [e.g. the definition in 10, Section 2.1], the common understanding of motion segmentation is that it is an *unsupervised* clustering process. The origin lies in the factorization of the motion of a moving affine camera observing a static scene: The measurement matrix **W**, which contains observed point positions over time forming trajectories, can be decomposed into the static 3D shape **S** of the scene and the camera motion **M** over time. Vice versa, a static camera may observe one rigid moving object and obtain its **M** and **S**. However, if a dynamic scene is observed, multiple motions occur rendering the decomposition impossible. The *segmentation* into independent motions finds the reconstruction of motion and shape of all objects *and* the assignment of trajectories to the objects.

Earlier work focused on finding subspaces in **W** and assigning the trajectories to them, e.g. with the generalized PCA [11]. Subsequent work exploited sparsity in the manifold of the observed point trajectories [12] or in features generated from them [13]. More recently, Shi et al. [14] integrated temporal smoothness of the trajectories into the optimization by a discrete cosine transform (DCT) representation. While these subspace-based approaches are mathematically elegant and show good results in benchmarks such as the Hopkins 155 [15], especially missing data poses a serious problem: Such data has to be interpolated prior to the segmentation which becomes infeasible once the sequences become long and missing data dominates, which occurs e.g. due to occlusions. An exception is the approach of [16] which base their subspace-based method on many two-view motion observations. Due to the nature of this algorithm, missing data can be replaced by zeros without biasing the final result.

In contrast to the subspace-based approaches, Brox, Fradet, and Dragon, [17–20] analyze pairwise or higher-order [21] relationships between trajectories which are then aggregated to an affinity matrix **A** encoding the trajectory similarity. A final spectral clustering [22] step of **A** finds the association of trajectories to motion segments. Brox and Malik [17] use similarity in translational motion and spatial distance as affinities to cluster trajectories obtained from particle trajectories. We follow several concepts of theirs, but use an energy-based formulation instead of affinities. Fradet [18] randomly samples more complex affine motion models and use J-Linkage [23] as final clustering of trajectories having many common inliers. Similarly, Dragon et al. [19,20] combine multiple motion hypotheses of frame-to-frame motion segmentation allowing complex motion models and real-time processing. We adopted their idea of using multiple time scales to propagate motion knowledge through occlusions. Since all these approaches only analyze data pairs instead of the subspaces formed from the complete data, missing data is not an intrinsic problem: If no affinity may be obtained for a trajectory pair, it is set to zero – the lowest possible affinity. Since spectral clustering only enforces high affinities to be in the same segment but does not penalize low affinities within one, it is especially suitable for this clustering problem. As alternative to the final clustering step, an energy-based optimization was proposed by Lezama et al. [24]. This allowed them to specify a static depth ordering constraint into the clustering.

In our approach, we integrate prior knowledge about the assignment of trajectories into motion segments. However, directly modifying **A** to prevent trajectories with the same prior label to be clustered into different segments has undesired biases which leads to poor results: Since spectral clustering minimizes the inter-segment cut through **A**, weighted by the sum of intra-segment weights [22], tweaking local weights creates a global bias which results in non-intuitive results.

Related to adding hard constraints to clustering is the field of *Transductive Learning* which is learning a classifier from partially labeled data with its only purpose being to classify the rest of the data. Regarding *Spectral Clustering*, Zhou and Burges [25] treat initial labeling as a second "view" on the data with a second graph which is combined with the regular affinities. However, the labeling constraints become soft constraints and setting the penalty for a label violation to a high value leads to an undefined optimization of the unlabeled data.

Another term for the same problem is *weakly supervised clustering*. Tuzel et al. [26], extended mean shift clustering to add hard labeling constraints by projecting the input data to a high-dimensional space in which all hard constraints are fulfilled. While this approach shows promising results and does not have the aforementioned weaknesses, the choice of the Kernel is critical but not intuitive.

In our work, we overcome the no-prior-labeling limitation by formulating the motion clustering task as a multi-label Markov random Field (MRF), similar to graph-cut-based image segmentation [27]. We use the generalized Potts model, thus encouraging large motion differences between segments and similar motions within segments. This may seem straightforward, but to the best of our knowledge such a scheme has only been proposed before by Lezama et al. [24]. They used an MRF to specify depth constraints, but initial labels were not incorporated.

The common video segmentation approaches use motion as an important prior, but in contrast to the motion segmentation paradigm, this prior is only evaluated on a frame-to-frame level as, e.g., by Sun et al. [28]. In contrast, we segment sparse motion information from trajectories which last over many frames. The semi-supervised approach of [29] makes use of an MRF with the optical flow and the class label as hidden variables. Unary energies are obtained from foreground and background priors, and binary energies enforce spatial and temporal smoothness wrt. the labels and the motion. Grundmann et al. [30] presented an approach for unsupervised video segmentation. They oversegment the video with an extension of Felzenzwalb's image segmentation approach [31] and merge these segments with agglomerative clustering until a pre-defined distance threshold is reached. While they also incorporate motion features from the optical flow, their approach does not use long-time motion information. In the same scenario, Zhang et al. [32] tackle the problem of extracting the primary object of a sequence by using an *objectness* detector [33] as prior to create multiple hypotheses. The best hypothesis is chosen from the best path through a directed acyclic graph having unary edges from objectness score and binary ones describing appearance and shape similarity of proposals from two successive frames. Although the results of these works seem convincing, the evaluation sequences, e.g. the GT-SegTrack database, only contain objects with big motion boundaries which do not occur in our sequences. Furthermore, they are one magnitude shorter (21–70 frames) than our sequences (1000 frames) and thus not suitable for our data.

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

## 3. Integrating prior knowledge into motion segmentation

Before going into details about how the prior knowledge is integrating to the motion segmentation, there should be some words about how the trajectories are developed. The trajectories are developed as proposed by Sundaram et al. [34]; first, LDOF [34] is used to obtain the flow fields. Then, flow fields are concatenated to develop trajectories, which are initialized by a grid of points located on the first frame. A trajectory could terminate because of occlusion or in case of fast motion, and new trajectories are developed if there is an area without any trajectory on it.

### 3.1. Energy formulation

This section describes the energy minimization framework used to segment the trajectories into separate groups. Let $\mathcal{G} = (\mathcal{S}, \mathcal{N})$ be an undirected graph where each vertex $s \in \mathcal{S}$ represents a trajectory, and each edge $(s, r) \in \mathcal{N}$ models the relation between neighboring trajectories. Each edge has a non-negative weight, a function of similarity of the vertices it connects. The motion segmentation is obtained through a multi-label graph-cut that minimizes an energy associated with $\mathcal{G}$. There are different ways to measure the quality of a segmentation. In general we want the elements within a segment to be similar, and elements between different segments to be dissimilar. This means that edges between two vertices in the same segment should have high weights, and edges between vertices in different segments should have lower weights.

For the set $\mathcal{S}$, an acceptable segmentation results in $n$ subsets $\mathcal{O}_i \subseteq \mathcal{S}$ such that $\mathcal{O}_i \cap \mathcal{O}_{j\neq i} = \emptyset$ and $\bigcup_{i\in\{1,\dots,n\}} \mathcal{O}_i = \mathcal{S}$. Similar to its use in an already extensive image segmentation literature [35–38], the cost of a cut can be defined with an energy functional of the following type:

$$E(L) = \sum_{s\in\mathcal{S}} D(s, L(s)) + \sum_{(s,r)\in\mathcal{N}} V(s, r, L(s), L(r)), \tag{1}$$

where $L : \mathcal{S} \to \{1, 2, \dots, n\}$ assigns a label to each trajectory in the set. The data term $D$ is a data penalty function that assesses the relevance of individual trajectory labels (see Eq. (9)), and $V$ is an interaction potential that assesses the consistency of pairs of trajectories (see Eq. (11)). The neighboring set $\mathcal{N}$ is defined as

$$\mathcal{N} = \{(s, r) | s, r \in \mathcal{S} \quad \text{and} \quad d_{\text{sp}}(s, r) \leq \epsilon\} \tag{2}$$

where $d_{\text{sp}}(s, r) = \frac{1}{n_c(s,r)} \sqrt{\sum_t |\mathbf{x}^s(t) - \mathbf{x}^r(t)|^2}$ is the average spatial Euclidean distance over the common frames of the two trajectories $\mathbf{x}^s$ and $\mathbf{x}^r$, with $n_c(s, r)$ the number of frames that $s$ and $r$ have in common. A large neighborhood, controlled by $\epsilon$, decreases homogeneity of the segments. If the neighborhood is too small, on the other hand, the chance of getting trapped in a local minimum is high. We empirically found the threshold $\epsilon = 10$ pixels to be an appropriate choice. Finally, the cut with minimum cost is obtained by minimizing this energy functional as proposed in [39].

#### 3.1.1. Trajectory similarity

Since trajectories belonging to the same object move similarly and also tend to be spatially closer than trajectories with different associations, the measure for trajectory similarities contain two factors. One includes motions information and the other spacial distances as follows:

$$d^2(t, s, r) = \bar{d}_{\text{sp}}(s, r) d^2_{\text{mot}}(t, s, r), \tag{3}$$

$d$, $\bar{d}_{\text{sp}}$ and $d_{\text{mot}}$ are the total, normalized spatial and normalized motion distances at time $t$, respectively, defined in the following.

Unlike [17], we normalize the spatial distance by a factor that counts the number of common frames (i.e. during which both trajectories exist):

$$\bar{d}_{\text{sp}}(s, r) = \frac{d_{\text{sp}}(s, r)}{\delta_c(s, r)}, \tag{4}$$

where $\delta_c(s, r) = \log(n_c(s, r) + 1)$. This takes into account that the more frames two trajectories have in common the more reliable the result is. Trajectories terminate in case of occlusion or fast motions. In any of these situations, the optical flow is likely to be temporarily inaccurate, affecting short trajectories much more than longer ones. In addition, longer trajectories are richer in their motions and thereby carry more information in regard to similarity. Therefore, it is reasonable to put more weight on similarities obtained over a longer run.

The normalized motion distance is defined as follows:

$$d^2_{\text{mot}}(t, s, r) = \frac{\|\mathbf{v}^s(t) - \mathbf{v}^r(t)\|^2}{5\sigma^2(t, s, r)}, \tag{5}$$

where $\mathbf{v}^s(t)$ is the aggregated motion of a trajectory $s$ over 5 frames $\mathbf{v}^s(t) = \mathbf{x}^s(t + 5) - \mathbf{x}^s(t)$. In this equation, $\sigma$ has to be defined such that it can deal with both fast and slow motions in an even-handed manner. In particular, local variations among velocities within a segment should be tolerated more if motions in the segment are changing more rapidly. Therefore, $\sigma$ is defined in a way similar to [17], as follows:

$$\sigma(t, s, r) = \min_{a\in\{s,r\}} \sum_{t'=0}^{4} \sigma_v(\mathbf{x}^a(t + t'), t + t'), \tag{6}$$

where $\sigma_v$ is the variation in a local flow field. In our case it is the standard deviation of the flow field in a 10 pixel $\times$ 10 pixel window centered on the trajectory at each frame. This assures that small changes in cases with slow motion has more importance than cases with fast motions.

As long as two objects move next to each other, they share similar motions, and it is impossible to separate them as different objects. But as soon as they start to move differently, they could be distinguished. In order to exploit this information, the similarity between two trajectories considers the instance when they are differ the most. So,

$$d(s, r) = \max_t d(t, s, r). \tag{7}$$

Finally, the trajectory similarity between two trajectories is defined as:

$$w(s, r) = \exp(-d^2(s, r)). \tag{8}$$

We set $w(s, r) = 0$ for trajectories that have no temporal overlap.

#### 3.1.2. Data term

Let $\mathcal{O}_{L(s)}$ be the set of trajectories that are initially assigned label $L(s)$. $\mathcal{S}_l = \cup_{i=1:n}\mathcal{O}_i$ is then the set of trajectories that are initially labeled. Let $I(s) \in \{1, 2, \dots, n\}$ be the true label of these trajectories. Then, the data term in Eq. (1) is defined as

$$D(s, L(s)) = \begin{cases} 0 & \text{if } s \in \mathcal{S}_l \wedge L(s) = I(s) \\ K(s) & \text{if } s \in \mathcal{S}_l \wedge L(s) \neq I(s) \\ g(s, L(s)) & \text{if } s \notin \mathcal{S}_l \end{cases} \tag{9}$$

where $K(s) = 1 + \sum_{r:(r,s)\in\mathcal{N}} V(s, r, L(s), L(r))$ is a large value that enforces trajectories which are initially labeled to preserve their labels during the optimization process. $g(s, L(s))$ is a measure of dissimilarity between trajectory $s$ and subset $\mathcal{O}_{L(s)}$.

The prior knowledge gives information about assignment of trajectories that are initially labeled. But, we also use the prior knowledge to assign a likelihood of belonging the other trajectories to each of the segments. To do so, the penalty of assigning a trajectory to a segment which its initially labeled trajectories are inconsistent with that trajectory should be high, and low otherwise. So, we define

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

$g(s, L(s))$ as negative log-likelihood of the average trajectory similarity $w(s, r)$:

$$g(s, L(s)) = -\log\left(\underset{r\in\mathcal{O}_{L(s)}}{\text{mean}}\,(w(s,r))^{\gamma}\right). \quad (10)$$

In this equation the operator *mean* is used to compute an average similarity between a trajectory and a set of labeled trajectories, it could be either arithmetic (what we used) or geometric. In addition, since $w$ takes values in [0, 1], the negative log is a common way of transforming such a similarity value to an energy value, as, e.g., in [17].

#### 3.1.3. Pairwise term

We define the pairwise term in Eq. (1) as:

$$V(s, r, L(s), L(r)) = q(L(s), L(r))\, f(w(s, r)) \quad (11)$$

where $f$ is a monotonically increasing penalty function of the trajectory similarity $w(s, r)$. $q(L(s), L(r))$ denotes the way trajectories are compared to each other based on their labels. We use

$$q(L(s), L(r)) = 1 - \delta(L(s), L(r)), \quad (12)$$

where $\delta$ is the Kronecker delta function. If two trajectories are assigned to the same segment, no pairwise penalty is considered in the energy functional (1), and if they are assigned to different segments a penalty equal to $f(w(s, r))$ is added.

Based on the definition of $q$, it can be inferred that $f$ must be a monotonically increasing function of $w(s, r)$ because it should penalize similar trajectories assigned different labels, and no penalty is due if they are assigned the same label. Given that $0 \le w(s, r) \le 1$, $f$ is defined as the negative log-likelihood of the counter probability of $w$ which is weighted by $\phi$:

$$f(w(s, r)) = -\log\left(1 - (w(s, r))^{\phi}\right). \quad (13)$$

If two trajectories are similar, $w(s, r) \to 1$ and thus $f(w(s, r)) \to \infty$. If they are dissimilar $w(s, r) \to 0$ and $f(w(s, r)) \to 0$. As a result, the optimization of 1 tries to put similar trajectories into the same segment. $\gamma$ and $\phi$ non-linearly balance $D$ and $V$ in Eq. (1). They are empirically set to $\gamma = 0.1$ and $\phi = 0.001$.

Due to occlusions and fast motions, trajectories are asynchronous and span different temporal windows. Considering just trajectories that last for the whole shot would lower the number of tracked points, leaving us possibly even with an empty set. Therefore we obtain the pairwise energy for all trajectories that have at least one frame in common. Due to transitivity, it can be expected that even trajectories that share no frames can still be paired [17].

It would be desirable to penalize intra-segment dissimilarity analogue to penalizing inter-segment similarity. However, due to non-submodularity, the overall energy becomes intractable and the results become worse.

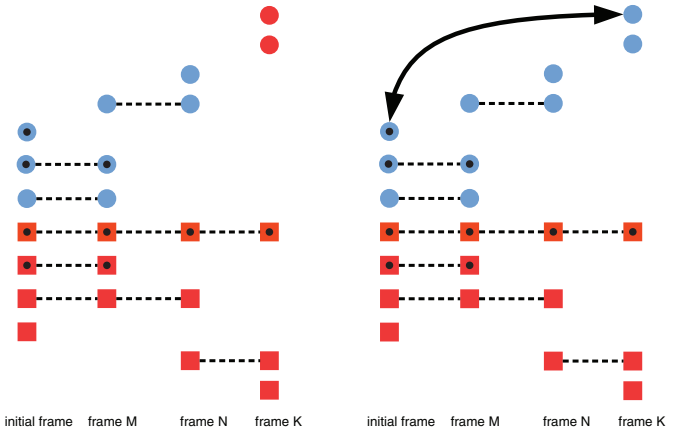
## 4. Particle matching

Similar to extensive graph-based image segmentation work [35–38], our motion segmentation method needs prior knowledge to base the segmentation upon. This prior knowledge is the assignments of a small subset of trajectories belonging to each segment. The true assignments of these trajectories are shared with the optimizer. Unlike image segmentation where the prior knowledge lasts during the optimization problem, a cumbersome problem with trajectory segmentation is that the initially labeled trajectories might not last for the whole shot. This happens in case of occlusions or a fast motions. So all trajectories of a segment may end; consequently, there is no trajectory left to represent that segment. Due to having no common frames, the data term defining the energy of assigning trajectories to that segment is very high for trajectories that are initialized from that moment on, while it is smaller for the other segments that have common frames with those trajectories. So, it is more probable that these trajectories would not be labeled as belonging to the terminated segment although that segment might be the right one.

To overcome this problem, in [9] we provided more prior knowledge by manually labeling trajectories in every 500th frames. In this section, a new method is presented to overcome the problem as well as fulfilling our ultimate goal which is to perform motion segmentation with as little user interactions as possible. For simplicity, we refer to our motion segmentation excluding particle matching by *basic moseg* [9], and for motion segmentation including particle matching by *improved moseg*.

### 4.1. Matching by multi-scale optical flow

Since trajectories belonging to an object are terminated due to fast motion or occlusion, our idea is to redetect the objects when reappearing. Fig. 1 demonstrates how redetecting the objects would improve the segmentation results. A synthetic example is created; each row represents a trajectory over time that starts at a frame and eventually may end at another frame. The trajectories belong to two different segments, the shapes show the true assignments while the colors are the decisions of the motion segmentation method. The ideal segmentation results in blue circles and red squares. As can be seen, as long as there are some manually labeled trajectories representing each of the segments at a frame, both methods end up with the correct assignments in that frame. Although in frame M where the circle segment loses all of it is manually labeled trajectories, the segmentation is still correct because there is a trajectory that has got correct assignment because of having common frames with the manually labeled ones (this could happen in case of partial occlusion). The main problem with the *basic moseg* is in situations where not only all the manually labeled trajectories for a segment are terminated, but there is no trajectory left that has common frames with them; for example, in case of complete occlusion. This happens in frame K where the *basic moseg* leads to a wrong segmentation. On the other hand, the *improved moseg* keeps up with the correct segmentation because it could find a particle in frame K that matches an initially labeled particle. In fact, its good performance is due to extending the




**(a)** *basicmoseg* **(b)** *improved moseg*

**Fig. 1.** Segmentation for a synthetic example. A set of trajectories belonging to two segments are shown, each row represents a trajectory over time that starts at a frame and may end at another frame. The shapes show the true assignments while the colors are the decisions of the motion segmentation, the ideal segmentation leads to blue circles and red squares. Those with black circle in the middle are the manually labeled trajectories. The double-sided arrow shows the matched particles by *improved moseg*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

manually labeled trajectories by finding new ones that match them.

*Improved moseg* tries to extend the prior knowledge by developing new trajectories that could last over occlusions. These trajectories are added to the trajectory set and since they are developed from a manually labeled frame, they get the true assignment and are used as prior knowledge. Then, segmentation is applied. These new trajectories are developed in two steps. First, objects are redetected by finding matched particles, then new trajectories are initialized for the matched particles. In order to redetect the objects, we obtain optical flow fields over multiple time scales. In other words, every $t$th frames is compared to the manually labeled frame and a flow field is calculated for each of them, matched points are found afterwards. In our experiments we set $t = 10$.

Let us suppose that the flow field from the labeled frame $t_0$ to frame $t$ is estimated, the next step is to derive matched points between these two frames. To do so, we initialize points in the labeled frame, and try to obtain matches for each of those points in frame $t$. To reduce the cost of matching and also because of unnecessary matching of points with no structure in their neighborhood, just points are initialized that show structure in their vicinity. To measure the structure, the structure tensor for a point in the image, $\mathbf{x}$, at each channel is obtained as follows:

$$\mathbf{S} = \mathbf{G} * \begin{bmatrix} \mathbf{I}_x^2 & \mathbf{I}_x\mathbf{I}_y \\ \mathbf{I}_x\mathbf{I}_y & \mathbf{I}_y^2 \end{bmatrix} \tag{14}$$

where $\mathbf{I}_x$ and $\mathbf{I}_y$ are the image derivatives in $x$ and $y$ directions, respectively, and $\mathbf{G}$ is a Gaussian kernel function centered on point $\mathbf{x}$ with standard deviation $\sigma = 2$. Then, the structure tensors of different channels are added up (here there are three channels, one for each color) to build the total structure tensor. Finally, points that second eigenvalue of their total structure tensor are smaller than a percent of average second eigenvalue for the whole image are removed from initialization.

The matches for initialized points are obtained by propagating each point in the labeled frame to frame $t$ using the relevant forward optical flow field $\mathbf{w}(t_0, \mathbf{x}(t_0)) := (u, v)^T$ via the following formula:

$$\hat{\mathbf{x}}(t) = \mathbf{x}(t_0) + \mathbf{w}(t_0, x(t_0)). \tag{15}$$

Where $\hat{\mathbf{x}}(t)$ is the match for $\mathbf{x}(t_0)$ in frame $t$.

This will find a match for each of the points in the labeled frame. But due to occlusion and possibility of wrong motion estimation in the optical flow field, unreliable matches must be removed. To do so, we use the backward flow field, the flow from $t$ to $t_0$. There are large inconsistency between the forward and backward flow nearby the optical flow boundaries and occluding area [40]. So, forward and backward flow fields are compared to each other. If they are not consistent for an specific point, either that point is occluded or the estimated flow field is not reliable. In either cases the match must be ignored. Let $\hat{\mathbf{w}}\big(t, \mathbf{x}(t)\big) := (\hat{u}, \hat{v})^T$ be the backward flow field, for a non-occluded point with $\mathbf{w}(t_0, \mathbf{x}(t_0)) = -\hat{\mathbf{w}}(t, \mathbf{x}(t_0) + \mathbf{w}(t_0, \mathbf{x}(t_0)))$. But because of inaccuracy in the flow estimation, we consider a tolerance bound such that if the difference of backward and forward flows exits this bound we ignore the match. As long as the following inequality is valid, a match could be valid:

$$|\mathbf{w} + \hat{\mathbf{w}}|^2 < 0.01(|\mathbf{w}|^2 + |\hat{\mathbf{w}}|^2) + 0.5, \tag{16}$$

as it is evident from this equation, the tolerance bound is proportional to the motion size for the subjected point, and the larger the motion, the more error is acceptable. We also ignore matches on the motion boundaries to prevent drifting. Therefore matches with

$$|\nabla u|^2 + |\nabla v|^2 > 0.01|\mathbf{w}|^2 + 0.002 \tag{17}$$

are deleted. This matching procedure is repeated for every $t$th frame from the manually labeled frame, and finally the match set is the set of all these matches.

To accomplish the second stage, that is to establish new trajectories on the matched points, we employ the same procedure as for the initialized points on the first frame. So, for each matched point on frame $t$ a new trajectory is created using the flow field of succeeding and preceding frames. Since the new trajectories are developed on matched point, they get the same label as their matches on the manually labeled frame, and therefore termination of initially labeled trajectories are compensated by creating these new trajectories.

## 5. Tracking based on segmentation

Along side segmentation, tracking is another important issue. Although there are many tracking algorithms already providing astonishing results on the type of sequences for which they have been designed for, we experimentally found none of them to perform sufficiently well on the cerebral palsy problem. The reasons are manifold: fast motions, high nonrigidity, frequent changes in appearance, etc. For example TLD [41], despite being fast and reliable for many applications, fails to track the limbs (upper two rows of Fig. 13). Therefore, we propose a motion segmentation based tracker, which tracks each of the objects using all trajectories belonging to the segment $\mathcal{O}$ related to that object. We could initialize our tracker manually, or by starting it from the center of mass of all trajectories in that segment at a labeled frame. Tracking using the location of the center of mass of $\mathcal{O}$ would fail due to discontinuity from partial occlusions. Instead, an iterative procedure is used to update the tracking results, as follows. For each segment $\mathcal{O}$ and each time step $t$, we define the subset of all trajectories $s \in \mathcal{O}$ that are visible at $t$ and $t+1$ as $\mathcal{S}(t)$. Let $\mathbf{x}^s(t)$ and $\mathbf{x}^s(t+1)$ denote the respective locations of $s$. Then, the tracking result $\mathbf{x}$ is updated iteratively using the average motion of the trajectories:

$$\mathbf{x}(t+1) = \mathbf{x}(t) + \frac{1}{|\mathcal{S}(t)|} \sum_{s \in \mathcal{S}(t)} (\mathbf{x}^s(t+1) - \mathbf{x}^s(t)). \tag{18}$$

Since Eq. (18) builds the update step by exploiting a large number of trajectories, it can filter out noise and unreliable trajectories, as long as their effects remain small compared to that of the majority of correctly labeled trajectories.

## 6. Experimental results and discussion

In all of the experiments in this section we used the dense optical flow proposed by Brox and Malik [17] to generate sparse trajectories with region density of 2.5%. Two different data sets are used to study the performance of our proposed methods. First, as the primary motivation for starting this work was to largely automate the assessment of infant general movements for the prediction of cerebral palsy, a set of infants' videos are studied in more details. Second, in order to investigate the applicability of our method, it is tested on a standard benchmark.

### 6.1. Performance on videos of infants

In all experiments in this section, we used the video set-up that was used in St. Olavs Hospital, Trondheim, Norway. During the experiments, we analyzed the first 1000 frames of 10 video sequences showing different infants carrying out different motions (Fig. 2). These sequences are a magnitude longer than the Hopkins 155 [15] and the Freiburg-Berkeley [42] dataset with an average length of 30 and 245 frames, respectively. As ground truth, we manually annotated a dense segmentation of every 250th frame as displayed in Fig. 3. Fig. 5 shows the average length of the trajectories of 10 sequences used in this study. As it can be seen, due to occlusions, fast and complicated motion patterns, the trajectories last just for 96.5 frames in average.

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

**Fig. 2.** Overview of the 10 sequences used in our experiments.

**Fig. 3.** Seq. 1 ground-truth segmentation for frames 1, 50, 200, 300 from left to right.

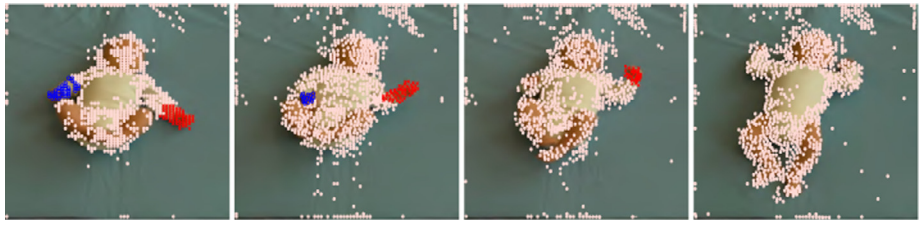


**Fig. 4.** Seq. 1 segmentation results of [17] for frames 1, 50, 200, 300 from left to right.

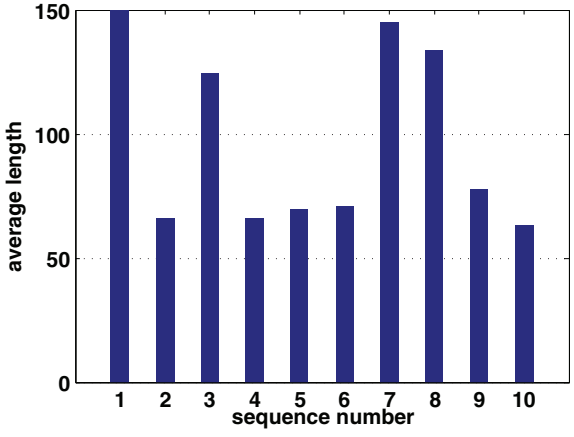



**Fig. 5.** Average length of the trajectories in number of frames for different sequences.

*6.1.1. Segmentation*

Fidgety Movements assessment is a strong cue for CP prediction. "Fidgety movements are defined as an ongoing stream of small, circular, and elegant movements of neck, trunk, and limbs in all directions" [43]. Based on this definition, it is crucial to capture the motions of the body parts in very fine details in order to predict CP. As it is visible from the ground truth (Fig. 3) the goal of segmentation in our application is to capture the motion of six different segments representing hands, feet, head and trunk.

Fig. 4 shows the segmentation results for [17] which is an unsupervised technique. It is not hard to discover that this method could separate only very distinct motions from each other, and most of the trajectories are assigned to the background. Its poor performance can hardly be criticized. Fast and complicated motion patterns of body parts makes the segmentation task challenging. Additionally, particles on the same desired segment move quite differently and without additional knowledge motion is not a strong enough cue to meet the segmentation demands. This additional knowledge could be provided as a set of prior knowledge carrying information about the correct assignment of a subset of trajectories representing the desired segments.

We integrate this prior knowledge to our segmentation algorithm by manually labeling a small subset of trajectories as shown in Fig. 7. For all experimental results in the followings, for each sequence *two* frames (1 and 500) are manually labeled and fed to the *basic moseg*, while there is only *one* manually labeled frame (the first frame) used in *improved moseg*. Frames 250 and 750 are considered for evaluation. Fig. 6 shows the rate of trajectories used as prior knowledge. As it indicates, 5% of the trajectories are being a priory labeled for the *basic moseg*, and just 2.6%, for *improved moseg*. Considering the segmentation difficulty of this application, these numbers are quite small.

Because our method is the first semi-supervised motion segmentation approach, for the sake of comparison, we use the prior labeling as naive baseline. We use the same prior knowledge as the *basic moseg* and without applying any segmentation method the results in 250th and 750th frames are compared with the ground truth. To obtain a measure of segmentation accuracy we calculate the F-measure between each segmented region $c_i$ and each ground truth region $g_i$ is defined as:

$$F_{i,j} := \frac{2|c_i \cap g_j|}{|c_i| + |g_j|}. \tag{19}$$

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

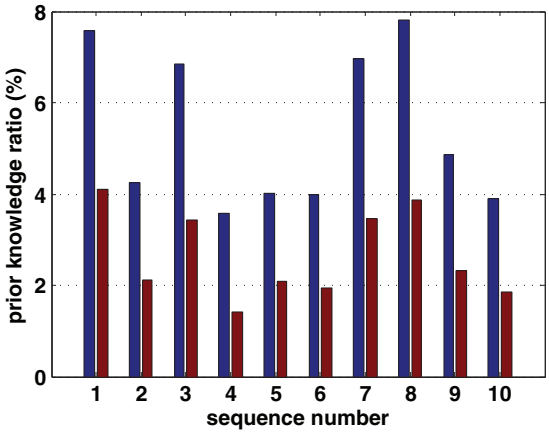



**Fig. 6.** Percentage of the trajectories used as prior knowledge for different sequences in blue for two frames labeled and brown for just the initial frame labeled. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**(a)** frame 1 **(b)** frame 500

**Fig. 7.** Initial labeling and the additional labeling at frame 500.

Where |.| denotes the size of each set. The average F-measure is shown for four cases in Fig. 8: *basic moseg* with one and two a priory labeled frames, the baseline, and the *improved moseg* with just one priory labeled frame. Poor results of the baseline indicates the level of complexity that the segmentation methods are dealing with. Despite being a challenging problem, our segmentation methods have managed to perform reliably. The *basic moseg* with one labeled frame as a priory has gained 77.62% segmentation accuracy, increasing the prior knowledge to two labeled frames boosts up the results to 94.92%. Furthermore, the improved moseg outperforms the basic one by obtaining 96.34% accuracy with using just one labeled frame as prior knowledge.

Fig. 9 shows a qualitative overview of our segmentation results for the 10 sequences, a more detailed overview can be found in Figs. 15d and 16. For a more profound study, segmentation results for one of the sequences are shown in the lower two rows of Fig. 10 alongside with results of the baseline in the upper two rows. It is clear that the baseline in itself performs poorly, while our method exhibits significantly better performance.

Occlusion is a longstanding problem in motion segmentation. Frames 650–800 of Fig. 10 shows a case of severe occlusion where the head is occluded by both hands. As it can be seen, the segmentation remains correct and in frame 950, the new trajectories in the occluded area on the head are labeled correctly. Partial occlusion is less of a problem for our proposed methods: there are some trajectories left that can still stand in for the terminated ones. These are joined by novel trajectories upon the reappearance of the previously occluded region. In case of a complete occlusion, trajectories could be linked to each other by providing further manual labeling for *basic moseg*, or by using *improved moseg* that propagates the initial labels.

In sequence 5 the infant rolls to the right side, therefore, many of a priory labeled trajectories terminated, this weakens the *basic moseg* performance. On the other hand, the *improved moseg* overcomes this problem by redetecting matched particle when the infant goes to a normal situation where all body parts are visible. This can be seen in Fig. 11 where at frame 550 during rolling to the side, the head and the right foot are occluded and the initially labeled trajectories for these segments are lost. Later on at frame 750 where the baby roles back to the normal situation, we could see that the *basic moseg* wrongly assigns the trajectories on the head and the left foot to the left hand and the left foot, respectively. However, the *improved moseg* performs a correct segmentation by re-matching the particle to those in the labeled frame.

### 6.1.2. Tracking

When it comes to human motion analysis it may be straight forward to apply pose estimation approaches, but for application of

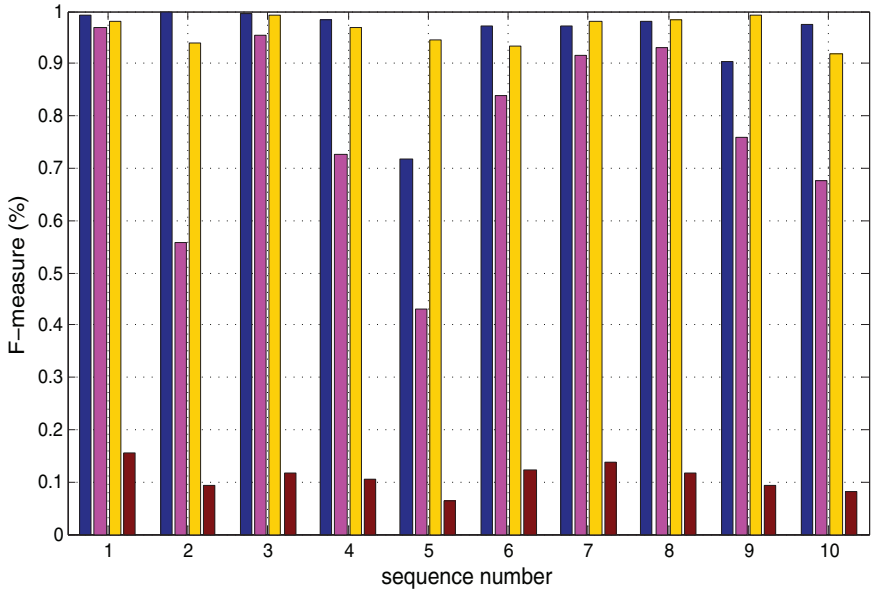



**Fig. 8.** The average F-measure for different sequences. Given are the results for the segmentation of *basic moseg* with one manually labeled frame in magenta, two manually labeled frames in blue, *improved moseg* with one labeled frame in yellow and for the baseline in brown. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

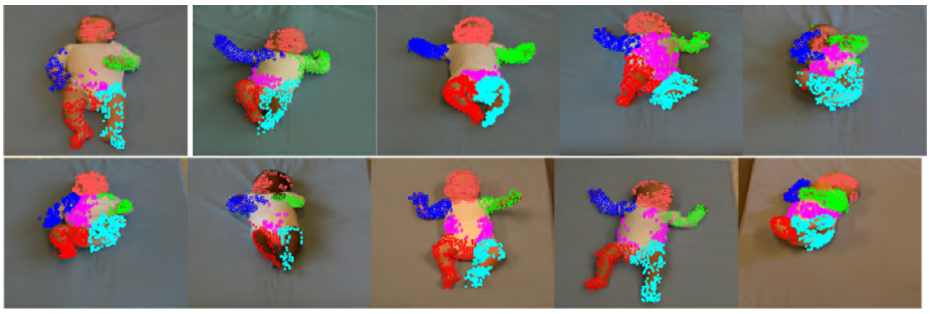


**Fig. 9.** Overview of the segmentation results of our *basic moseg* for all sequences in frame 250. ○, +, ×, ∗, □ and ◊ represent trajectories belonging to right hand, left hand, right leg, left leg, head and trunk, respectively.

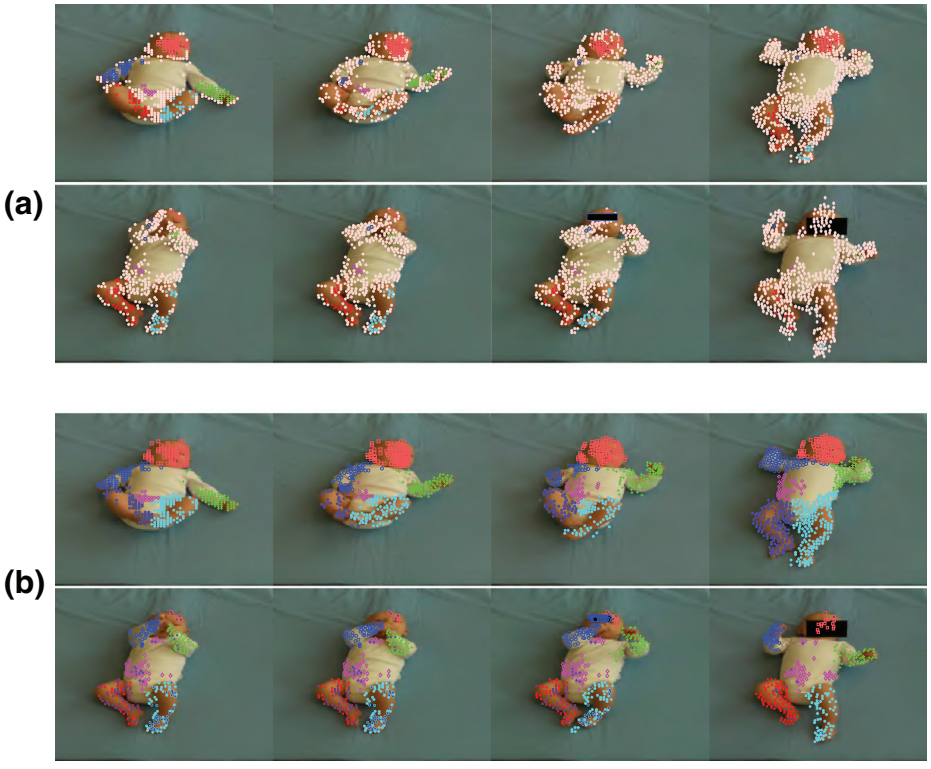



**Fig. 10.** Seq. 1 segmentation results for frames 1, 50, 200, 300, 650, 700, 800 and 950 from top left to down right. (a) shows the results for the baseline where no segmentation method is applied, and (b) is the results for the *improved moseg* method. Frames 800 and 950 have been anonymized after the segmentation. ○, +, ×, ∗, □ and ◊ represent trajectories belonging to right hand, left hand, right leg, left leg, head and trunk, respectively. ∗ in the first row shows trajectories which are not assigned any labels.

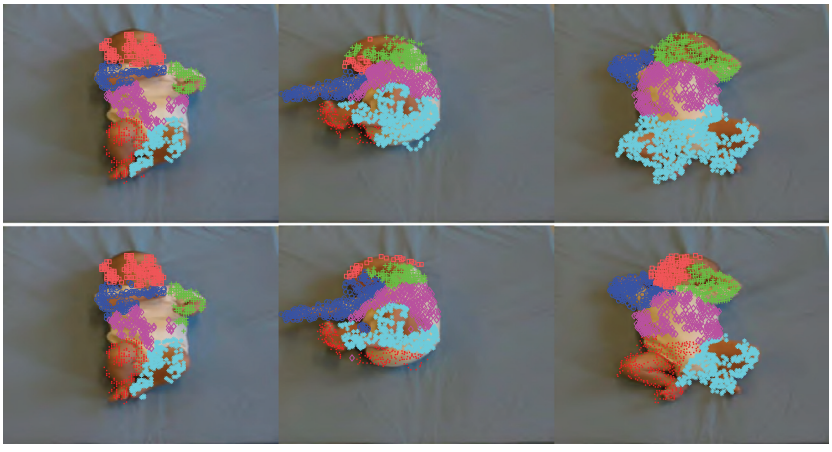


**Fig. 11.** Segmentation results of sequence 5 in frames 1, 550 and 750. The upper row is for the *basic mosec* with frames 1 and 500 as prior knowledge, and the lower row for *improved moseg* with only frame 1 as prior knowledge.

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

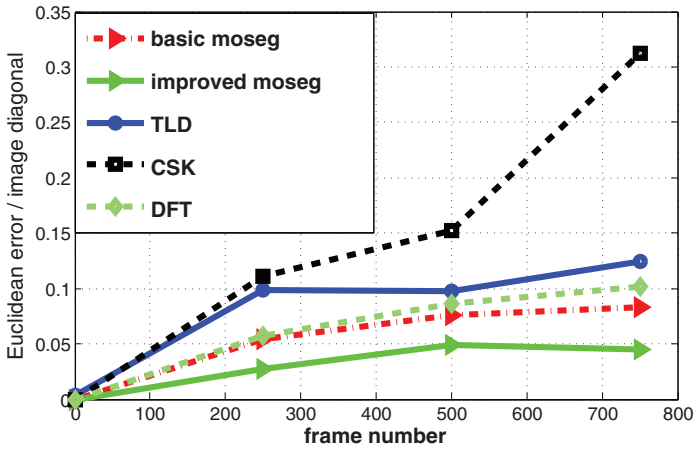



**Fig. 12.** The normalized Euclidean tracking error of *basic moseg*, *improved moseg*, TLD, CSK, and DFT. Results are the average over all six body parts and all 10 sequences. All errors are normalize with the size of the images.

**Table 1**
Ratio of average number of frames with no tracking over all sequences and segments.

| TLD | *Basic moseg* | *Improved moseg* | CSK | DFT |
|---|---|---|---|---|
| 12.46% | 0.38% | 0% | 0% | 0% |

CP detection it is important to capture the body parts motions in very fine detail. Because pose estimation methods have skeleton constraints as additional prior while our method has the same input as a tracker, we compare the performance of our method with tracking approaches. Our tracking performance is compared with TLD [41], CSK [44], and DFT [45] as representative for state-of-the-art video trackers. The ground truth information is provided for all six body parts for the 1st, 250th, 500th, and 750th frames of each sequence.

Fig. 12 plots the spacial error for each of the trackers over time. As quality metric, we calculate the Euclidean error from the ground truth segments center and average it over all body parts and all the

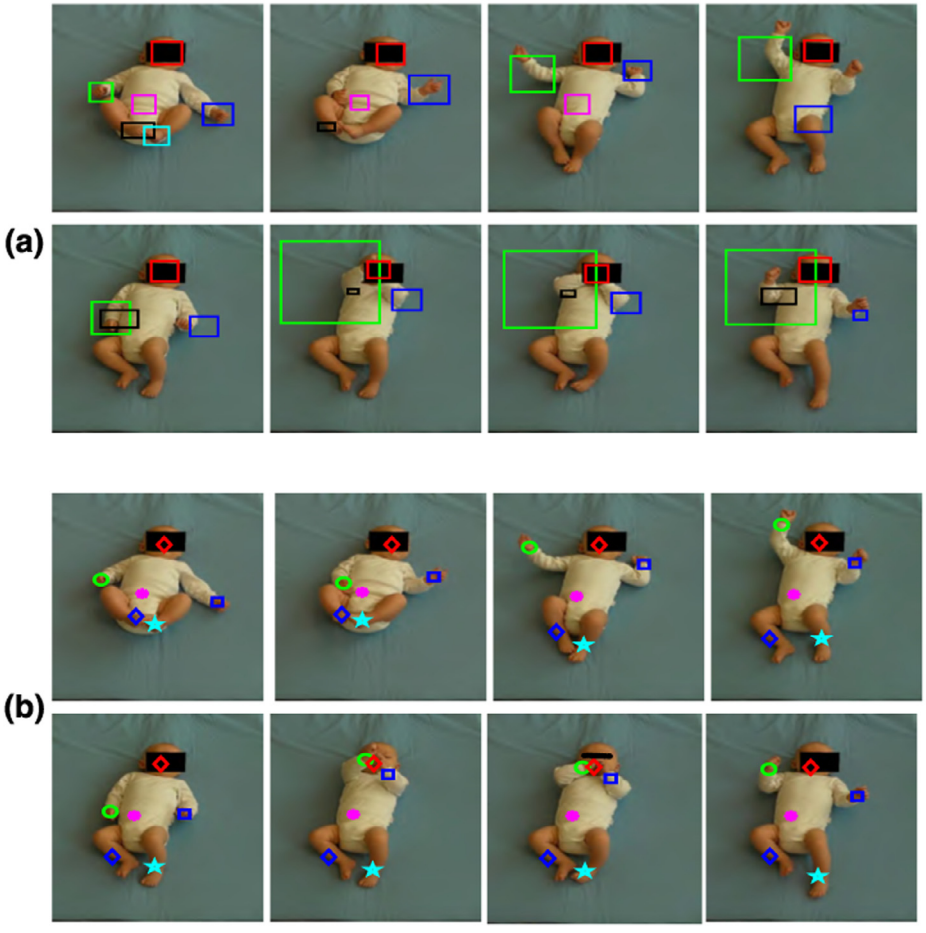



**Fig. 13.** Seq. 1 tracking results of TLD (a) and of proposed method (b) for frames 1, 50, 250, 350, 450, 650, 750 and 950 from top left to down right. □,□,□,□,□ and □ in the upper row, and ○, □, ◊, □, ★, ◊ and ∗ in the lower row represent track for right hand, left hand, right leg, left leg, head and trunk, respectively.

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

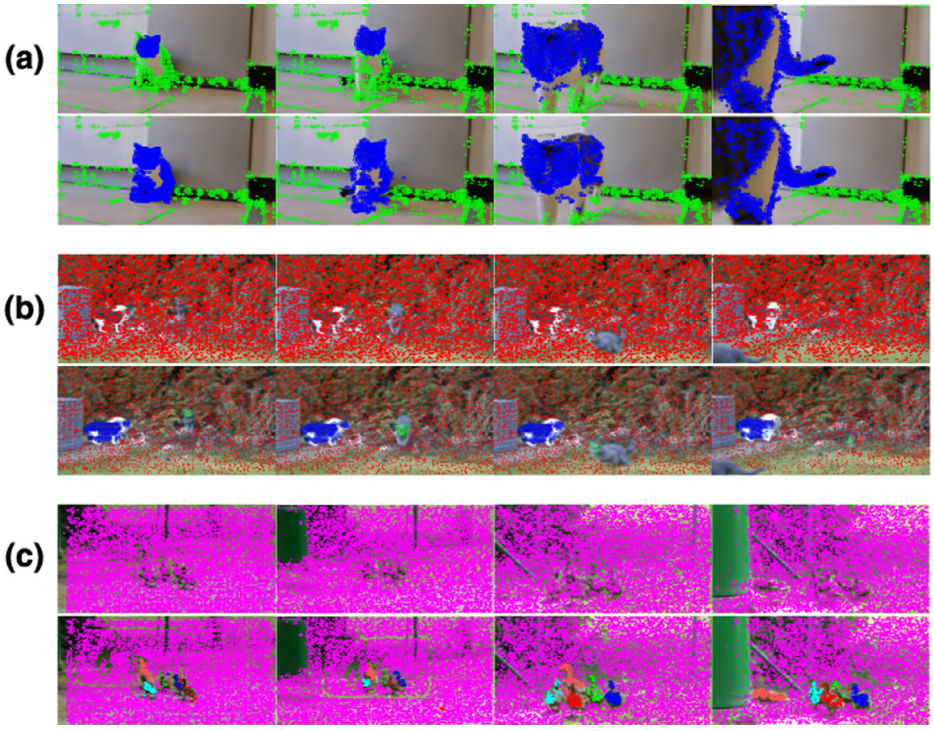



**Fig. 14.** Segmentation results for [17] (upper rows) and our *improved moseg* (lower rows) for frames (a) 1, 70, 90, 110 of *cats02*, (b) 1, 40, 50, 70 of *cats04*, and (c) 1, 100, 300, 380 of *ducks01*. For the sake of visibility, the background trajectories are thinned out in *cats04*.

10 sequences. All the trackers are initialized on the ground truth points in the first frame and run over the remaining frames. As it can be observed, all the trackers have an increasing error over time however our *moseg* trackers have the best performances. The tracker based on *improved moseg* reduced the error significantly, which is due to equipping the basic segmentation with particle matching.

A problem with many trackers is that they lose the object and therefore either they do not report any measurement or they falsely produce some results. For example TLD is one of the trackers that loses the track of the body parts frequently. Table 1 shows the percentage of frames where there is no detection by the tracker. It is clear that TLD has lost tracking quite often. Our *basic moseg* also lost tracking for a short period, it happened just for one of the sequences where during rolling to the sides one of the segments was completely lost. This problem is solved in the *improved moseg* because the particles are rematched when the infant goes to the normal situation (Fig. 11). The second case, where there is wrong detection, could even lead to worse errors for the CP detection. This is less of a problem for our proposed trackers because they are derived from a segmentation and therefore always update their results from the right object even though they might not report the right location.

For the sake of a qualitative visualization we plot the tracking results of TLD and our method in Fig. 13. The upper row shows a qualitative result of TLD. The tracker lost the left foot in frame 50. A bit later, the same happens to the right foot and in frame 450 TLD redetects it wrongly at the right hand. Similar problems regularly occur for the other body parts, hence the performance is insufficient for our task. The results of our proposed *basic moseg* tracker are shown in the bottom row of Fig. 13. All body parts are tracked efficiently in all the frames, without drifting or part loss. Occlusions (frames 650–750) were dealt with well.

### 6.2. Performance on cerebral palsy detection

In [8], we used our proposed motion segmentation and tracking approach in a larger scale. In that study we extract the motion data for 78 videos with the lengths of 3–5 min. It is worth mentioning that this is one of the largest data sets reported in the literature for CP detection. The extracted motion data are compared with data captured by electromagnetic sensors placed on the six different points that we aimed at. From both of the data sets a set of features are extracted and fed to an SVM classifier. The 87% accuracy of our data versus 85% of the electromagnetic sensors indicates the practicality of our method. Considering the facts that previous computer based approaches suffer either from being intrusive to the infant's motion pattern by using extra instruments, or lack precise analytic explanation for their results in one hand, and economically suitability of our approach in another hand, it is not far from expectation to say that our method has the potential of being widespread for clinical use.

**Table 2**
Segmentation results on the test set of the *Freiburg-Berkeley* data set.

| Method | Region density (%) | Precision (%) | Recall (%) | F-measure (%) |
|---|---|---|---|---|
| *Basic moseg* | 2.5 | $82.23 \pm 5.24$ | $67.45 \pm 3.92$ | $71.62 \pm 3.49$ |
| *Improved moseg* | 2.5 | $86.69 \pm 6.94$ | $70.85 \pm 4.83$ | $76.57 \pm 4.43$ |
| [17] | 2.5 | 76.75 | 60.38 | 65.05 |

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

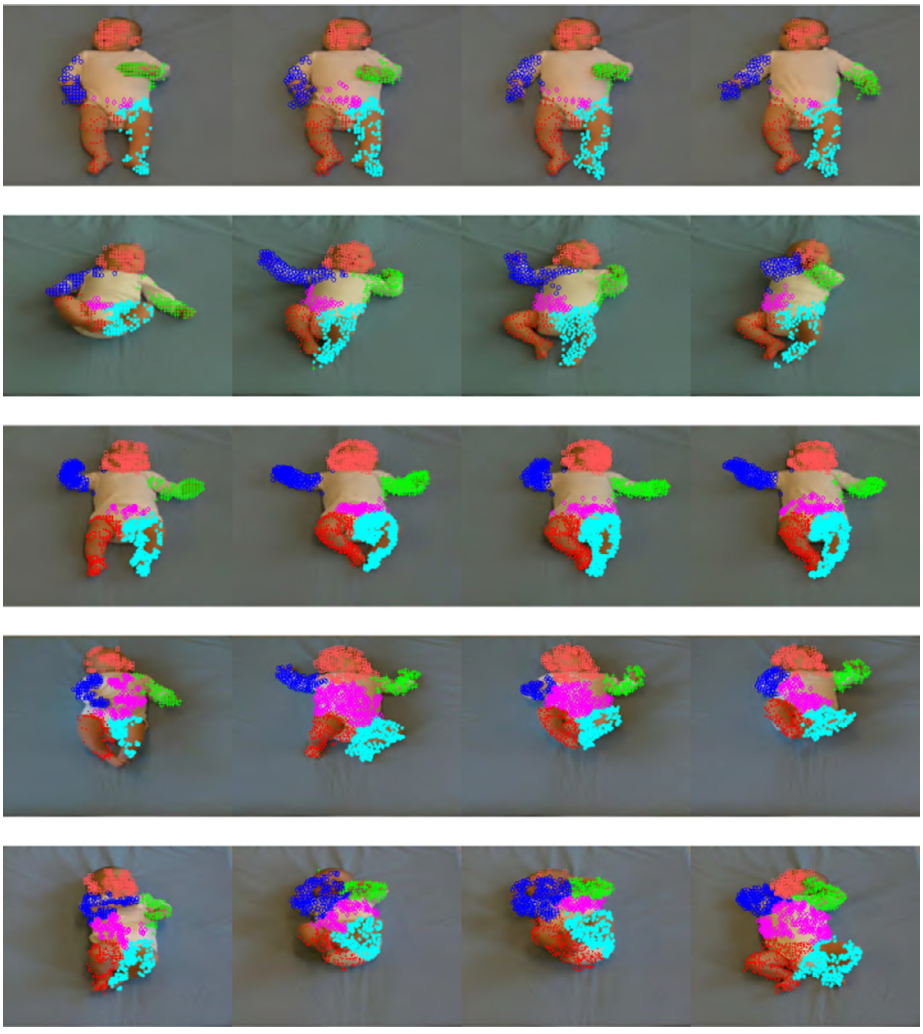


**Fig. 15.** Overview of the improved moseg results for sequences used in the paper. Rows show the results in frames 1, 250, 500 and 750 for the sequences 1–5 while only the first frame is used as prior knowledge.

### 6.3. Comparison with a standard benchmarks

In this section we will study how our methods perform on a different data type, two experiments are adopted. For the first experiment, the three video sequences *cats02*, *cats04* and *ducks01* of the *Freiburg-Berkeley* training data set [17,42] are considered, in which we deal with occlusion, disocclusion, camera motion, fast motion and low texture objects. The segmentation results of our proposed methods (initial labels in frame 50 in *cats02* and *cats04*, and 1 and 200 for *basic moseg* and 200 for *improved moseg* in *ducks01*) as well as those of Brox and Malik [17] are displayed in Fig. 14. As the results show, Brox and Malik [17] only distinguishes very different motions from each other. This is why no object is detected in Seq. *cats04* and *ducks01*. On the other hand, our segmentation method performs reliably: in Seq. *cats02* except for very small parts of the legs in a short period of the video, the cat is correctly segmented from the background. Seq. *cats04* shows a case where one of the cats has very low texture as well as fast motions, however the segmentation results are mostly correct. Finally, Seq. *ducks01* shows multiple similar objects that move next to each other, the segmentation task has become even more challenging because of occlusion, disocclusion and exit of one of the ducks. Despite all these, our method managed to segment all the objects correctly through the whole shot. It should be noticed that in *cats04* one of the cats has a low texture and almost no trajectory is initialized on it, this deteriorates the quality of our methods.

For the second experiment, we consider the whole test-set of the *Freiburg-Berkeley* data set [17,42] which contains 30 videos of different moving objects. The goal of this study is first to make the results comparable with prior and future published numbers, and second, is to study the sensitivity of the segmentation results to the frame that is manually labeled and used as prior knowledge. For each of the videos, at most 10 frames, some of the videos have less than 10 ground truth frames, from the ground truth frames that have the largest number of objects are randomly selected. Each time one of these frames is used to develop the prior knowledge while the whole ground truth frames are used for evaluating the segmentation results. The standard deviations of the comparison measurements show the sensitivity of our methods to selecting different frames as prior knowledge. Precision, recall, and F-measure introduced in [42] are used to measure the performance of each of the segmentation methods. The average over all 30 videos of these measurement alongside the average standard deviation of our methods by changing the prior knowledge are shown in Table 2. As it can be seen both of our methods outperform the method of [17], and *improved moseg* has the best performance. However, the performance of our methods on this data set is not as sufficiently good as the performance on the infants video data. The reason is mainly because

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

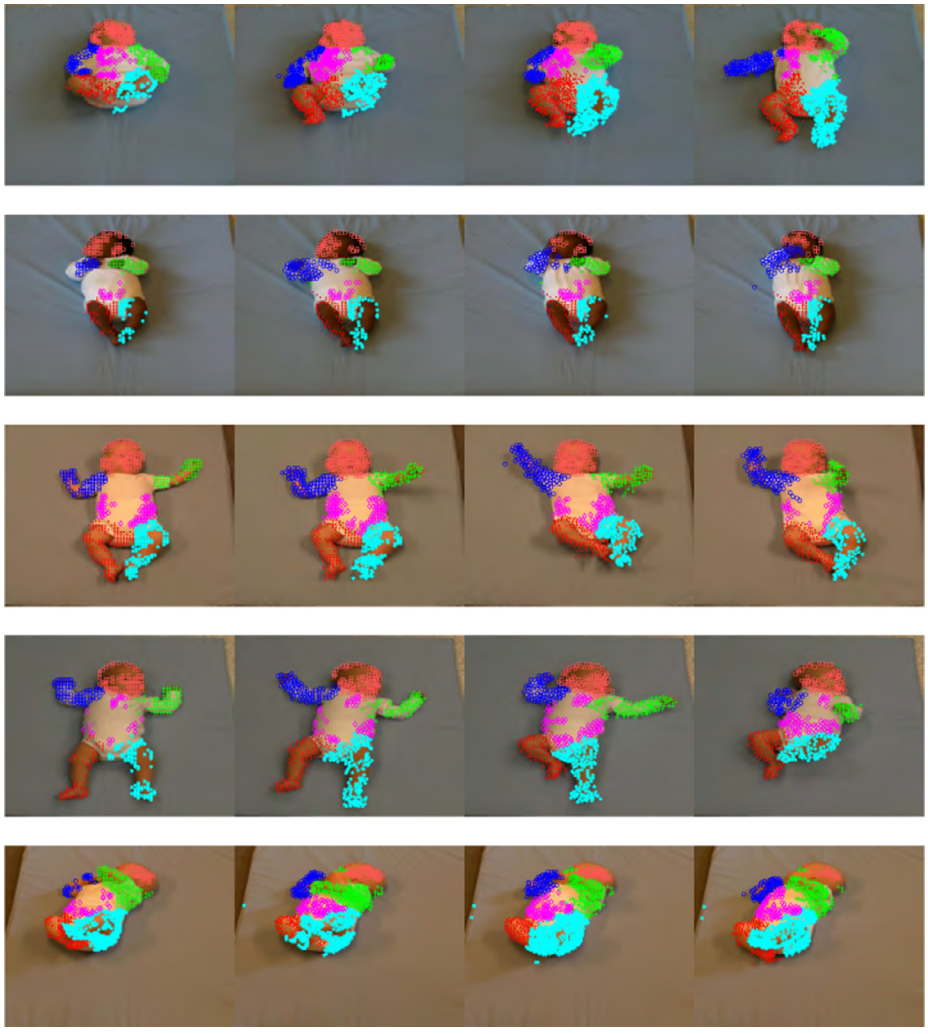


**Fig. 16.** Overview of the improved moseg results for sequences used in the paper. Rows show the results in frames 1, 250, 500 and 750 for the sequences 6–10 while only the first frame is used as prior knowledge.

our segmentation methods depend on the prior knowledge that is prepared for each objects. In this data set, complete occlusion, exiting and entering new objects, and camera shaking reduce the quality of the flow fields and, consequently, reducing the quality of our methods.

Although our methods perform reliably, they could suffer from some points. First, the segmentation task depends on the distance between trajectories, and since our distance measure (3) only allows for the verification of translational model, we might have problems with segmenting other models of motion. For example, the legs of the cat in Seq. *cats02* are wrongly segmented because we deal with fast scaling. The second problem could arise from optical flow or trajectory inaccuracy. Although we used one of the promising optical flow methods, we could still have problem in situations with fast motion and low texture. This could be visualized in the last image of Seq. *cats04* where the low texture cat does a fast jump and trajectories are wrongly developed.

Success of our methods depends on prior knowledge about the objects. Since providing prior knowledge is not possible or may be expensive for applications where objects frequently enter and exit the scene, our methods may not be favorable for a general segmentation purpose. Our methods are suited best for situations where precise segmentation is needed although at the expense of some prior knowledge.

## 7. Conclusions

In this paper, we dealt with motion based object segmentation and tracking in a video. Due to insufficiency of motion in situations where objects share similar motion pattern, need for additional information seems inevitable. We proposed a framework to integrate this additional knowledge to the segmentation procedure. The knowledge in our case was a set of assigned trajectories to the true segments, that was prepared by manually labeling some of the frames. We reduced the number of manually labeled frames by integrating a multi-scale particle matching technique to our method. Since the original motive of our work was to extract motion data for predicting cerebral palsy, detailed analytical experiments were adopted on videos of infants in order to see the functionality of our methods for this application. In addition, a standard data set was considered to show the generality of the methods. Quantitative and qualitative results confirmed that the proposed methods boosted the segmentation performance a great deal. Furthermore, the tracker derived from our segmentation methods outperformed the state-of-the-art tracking methods.

## References

[1] L. Meinecke, N. Breitbach-Faller, C. Bartz, R. Damen, G. Rau, C. Disselhorst-Klug, Movement analysis in the early detection of newborns at risk for developing spasticity due to infantile cerebral palsy, Hum. Mov. Sci. 25 (2) (2006) 125–144.

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004

[2] D. Karch, K. Wochner, K. Kim, H. Philippi, M. Hadders-Algra, J. Pietz, H. Dickhaus, Quantitative score for the evaluation of kinematic recordings in neuropediatric diagnostics, Methods Inf. Med. 51 (4) (2012) 341–347.
[3] D. Karch, K.-S. Kang, K. Wochner, H. Philippi, M. Hadders-Algra, J. Pietz, H. Dickhaus, Kinematic assessment of stereotypy in spontaneous movements in infants, Gait Posture 36 (2) (2012) 307–311.
[4] L. Adde, J.L. Helbostad, A.R. Jensenius, G. Taraldsen, R. Støen, Using computer-based video analysis in the study of fidgety movements, Early Hum. Dev. 85 (9) (2009) 541–547.
[5] L. Adde, J.L. Helbostad, A.R. Jensenius, G. Taraldsen, K.H. Grunewaldt, R. Støen, Early prediction of cerebral palsy by computer-based video analysis of general movements: a feasibility study, Dev. Med. Child Neurol. 52 (8) (2010) 773–778.
[6] L. Adde, J. Helbostad, A.R. Jensenius, M. Langaas, R. Støen, Identification of fidgety movements and prediction of CP by the use of computer-based video analysis is more accurate when based on two video recordings, Physiother. Theory Pract. 29 (6) (2013) 469–475.
[7] A. Stahl, C. Schellewald, Ø. Stavdahl, O.M. Aamo, L. Adde, H. Kirkerod, An optical flow-based method to predict infantile cerebral palsy, IEEE Trans. Neural Syst. Rehabil. Eng. 20 (4) (2012) 605–614.
[8] H. Rahmati, O.M. Aamo, Ø. Stavdahl, R. Dragon, L. Adde, Video-based early cerebral palsy prediction using motion segmentation, in: Engineering in Medicine and Biology Society (EMBC), 36th Annual International Conference of the IEEE, IEEE, 2014.
[9] H. Rahmati, R. Dragon, O.M. Aamo, L. Van Gool, L. Adde, Motion segmentation with weak labeling priors, in: Pattern Recognition, Springer, 2014, pp. 159–171.
[10] R. Jain, R. Kasturi, B.G. Schunck, Machine Vision, McGraw-Hill, 1995.
[11] R. Vidal, R. Hartley, Motion segmentation with missing data using powerfactorization and GPCA, in: Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 2, IEEE, 2004, pp. II–310.
[12] E. Elhamifar, R. Vidal, Sparse subspace clustering, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 2790–2797.
[13] A.M. Cheriyadat, R.J. Radke, Non-negative matrix factorization of partial track data for motion segmentation, in: Computer Vision, IEEE 12th International Conference on, IEEE, 2009, pp. 865–872.
[14] F. Shi, Z. Zhou, J. Xiao, W. Wu, Robust trajectory clustering for motion segmentation, in: Computer Vision (ICCV), IEEE International Conference on, IEEE, 2013, pp. 3088–3095.
[15] R. Tron, R. Vidal, A benchmark for the comparison of 3-D motion segmentation algorithms, in: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE, 2007, pp. 1–8.
[16] Z. Li, J. Guo, L.-F. Cheong, S.Z. Zhou, Perspective motion segmentation via collaborative clustering, in: Computer Vision (ICCV), IEEE International Conference on, IEEE, 2013, pp. 1369–1376.
[17] T. Brox, J. Malik, Object segmentation by long term analysis of point trajectories, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 282–295.
[18] M. Fradet, Contributions la segmentation de squences d'images au sens du mouvement dans un contexte semi-automatique, Ph.D. thesis, universit de Rennes 1, Mention Traitement du Signal et Telecommunications, 2010.
[19] R. Dragon, B. Rosenhahn, J. Ostermann, Multi-scale clustering of frame-to-frame correspondences for motion segmentation, in: Computer Vision–ECCV, Springer, 2012, pp. 445–458.
[20] R. Dragon, J. Ostermann, L. Van Gool, Robust realtime motion-split-and-merge for motion segmentation, in: Pattern Recognition, Springer, 2013, pp. 425–434.
[21] P. Ochs, T. Brox, Higher order motion models and spectral clustering, in: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, IEEE, 2012, pp. 614–621.
[22] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: NIPS, vol. 14, 2002, pp. 849–856.
[23] R. Toldo, A. Fusiello, Robust multiple structures estimation with j-linkage, in: Computer Vision–ECCV 2008, Springer, 2008, pp. 537–547.
[24] J. Lezama, K. Alahari, J. Sivic, I. Laptev, Track to the future: spatio-temporal video segmentation with long-range motion cues, in: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE, 2011.
[25] D. Zhou, C.J. Burges, Spectral clustering and transductive learning with multiple views, in: Proceedings of the 24th International Conference on Machine learning, ACM, 2007, pp. 1159–1166.
[26] O. Tuzel, F. Porikli, P. Meer, Kernel methods for weakly supervised mean shift clustering, in: Computer Vision, IEEE 12th International Conference on, IEEE, 2009, pp. 48–55.
[27] Y. Boykov, G. Funka-Lea, Graph cuts and efficient N-D image segmentation, Intl. J. Comput. Vis. 70 (2) (2006) 109–131.
[28] D. Sun, E.B. Sudderth, M.J. Black, Layered segmentation and optical flow estimation over time, in: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, IEEE, 2012, pp. 1768–1775.
[29] D. Tsai, M. Flagg, A. Nakazawa, J.M. Rehg, Motion coherent tracking using multi-label mrf optimization, ICVV 100 (2012) 190–202.
[30] M. Grundmann, V. Kwatra, M. Han, I. Essa, Efficient hierarchical graph based video segmentation, in: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, 2010.
[31] P.F. Felzenszwalb, D.P. Huttenlocher, Efficient graph-based image segmentation, Intl. J. Comput. Vis. 59 (2) (2004).
[32] D. Zhang, O. Javed, M. Shah, Video object segmentation through spatially accurate and temporally dense extraction of primary object regions, in: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, IEEE, 2013, pp. 628–635.
[33] I. Endres, D. Hoiem, Category independent object proposals, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 575–588.
[34] N. Sundaram, T. Brox, K. Keutzer, Dense point trajectories by GPU-accelerated large displacement optical flow, in: Computer Vision–ECCV 2010, Springer, 2010, pp. 438–451.
[35] D. Greig, B. Porteous, A.H. Seheult, Exact maximum a posteriori estimation for binary images, J. R. Stat. Soc. Ser. B (Methodol.) (1989) 271–279.
[36] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, Pattern Anal. Mach. Intell. IEEE Trans. 23 (11) (2001) 1222–1239.
[37] V. Kolmogorov, R. Zabin, What energy functions can be minimized via graph cuts? Pattern Anal. Mach. Intell. IEEE Trans. 26 (2) (2004) 147–159.
[38] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, Pattern Anal. Mach. Intell. IEEE Trans. 26 (9) (2004) 1124–1137.
[39] V. Kolmogorov, Convergent tree-reweighted message passing for energy minimization, Pattern Anal. Mach. Intell. IEEE Trans. 28 (10) (2006) 1568–1583.
[40] M. Proesmans, L. Van Gool, E. Pauwels, A. Oosterlinck, Determination of optical flow and its discontinuities using non-linear diffusion, in: Computer Vision ECCV'94, Springer, 1994, pp. 294–304.
[41] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-learning-detection, TPAMI 34 (7) (2012) 1409–1422.
[42] P. Ochs, J. Malik, T. Brox, Segmentation of moving objects by long term video analysis, Pattern Anal. Mach. Intell. IEEE Trans. 36 (6) (2014) 1187–1200.
[43] H.F. Prechtl, C. Einspieler, G. Cioni, A.F. Bos, F. Ferrari, D. Sontheimer, An early marker for neurological deficits after perinatal brain lesions, Lancet 349 (9062) (1997) 1361–1363.
[44] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, Exploiting the circulant structure of tracking-by-detection with kernels, in: Computer Vision–ECCV 2012, Springer, 2012, pp. 702–715.
[45] L. Sevilla-Lara, E. Learned-Miller, Distribution fields for tracking, in: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on, IEEE, 2012, pp. 1910–1917.

Please cite this article as: H. Rahmati et al., Weakly supervised motion segmentation with particle matching, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.07.004