Data-Driven Motion Pattern Segmentation in a Crowded Environments

Jana Trojanová[2(✉)], Karel Křehnáč[1], and François Brémond[2]

[1] Neovision, Prague, Czech Republic
[2] STARS, Inria Sophia Antipolis, Sophia Antipolis, France
jana.trojanova@inria.fr

Abstract. Motion is a strong clue for unsupervised grouping of individuals in a crowded environment. We show that collective motion in the crowd can be discovered by temporal analysis of points trajectories. First k-NN graph is constructed to represent the topological structure of point trajectories detected in crowd. Then the data-driven graph segmentation and clustering helps to reveal the interaction of individuals even when mixed motion is presented in data. The method was evaluated against the latest state-of-the-art methods and achieved better performance by more than 20 %.

1 Introduction

Understanding crowd dynamics in complex environments remains an open problem in computer vision due to a large number of individuals exhibiting diverse movement. The crowd dynamics evolve depending on each individual's will as well as that of his neighbors. The variety of interactions among individuals is what makes the task of crowd understanding a difficult problem. Some individuals can exhibit an aggregated motion while others can move independently. Figure 1 shows an example of motion pattern segmentation interpreting these interactions. The term *motion pattern* here represents a spatial region with coherent flow in comparison to its neighboring regions.

Prior research in motion pattern segmentation can mostly be classified into two groups: flow field model-based segmentation and similarity model-based clustering. The first one simulates a moving crowd as a time dependent flow field. The flow field consists of regions with quantitatively different dynamics and motion patterns emerge from the spatio-temporal interactions of individuals. They build on optical flow alike features and use methods such as edge-based segmentation, graph-based segmentation or watershed segmentation. These methods can well describe the structured crowd motion and are the most studied in the motion pattern segmentation. However, they are temporally inconsistent over a longer video shot and work only for high crowd density otherwise the video scene would be over-segmented. A nice overview of such methods can be found in [1].

The second group of methods uses the principle of clustering. Once low-level motion features are detected they are grouped based on some similarity

G. Hua and H. Jégou (Eds.): ECCV 2016 Workshops, Part II, LNCS 9914, pp. 760–774, 2016.
DOI: 10.1007/978-3-319-48881-3_53

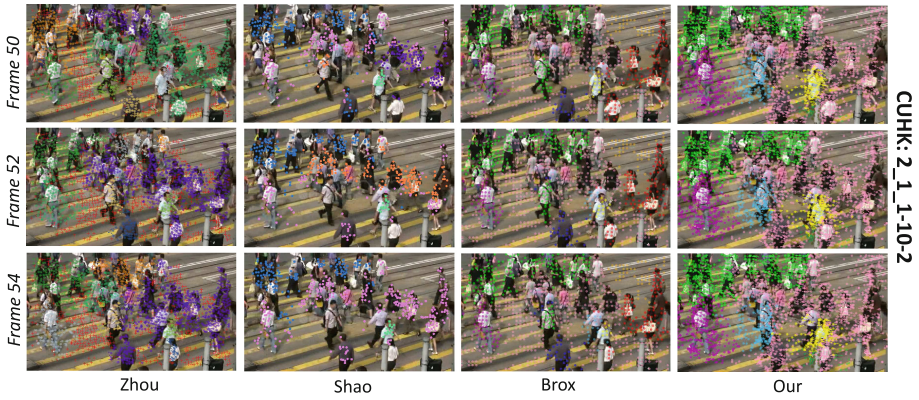

Fig. 1. Segmented motion patterns (or groups segregation) as the crowd evolves in time. The color of point represents the assignment to a group. While groupings are temporally consistent over a set of frames for Brox [2] and Our method. Zhou [3] and Shao [4] fail to maintain consistent groupings between frames. (Color figure online)

measurements or fitted with a probability model. The boom with similarity-based clustering is linked with the success of local motion features such as short point trajectories which can be obtained more easily than the whole trajectory. These points trajectories are more discriminative than local optical flows. The similarity clustering methods can handle structured and unstructured crowds. Representatives of such methods are [5,6]. Recently, probability models showed high potential in discovering semantic regions. Even though the methods can well capture overlapping behaviors and spatial dependencies among them, they require the whole video in advance to learn the probability model [4]. As will be shown in the experiment section the temporal consistency over a longer video shot is not always maintained.

Our framework builds on principles of both groups. No training is required and motion patterns are revealed by temporal analysis of point trajectories detected on a set of 15 frames. First of all an oriented graph is constructed based on k-nearest neighbor of points trajectories. Each node of the graph represents an averaged trajectory. The objective is to keep only the compact neighbor nodes with high collective motion. To do so the graph edges are weighted by correlation among two connected nodes. The graph is segmented by a single threshold to keep nodes with high collective motion. This results in small compact neighbor groups. Clustering is applied to merge compact neighbor groups with similar motion. The propagation of similar motion through neighbors allow us to discover coherent motion in the whole crowded environment. Figure 1 illustrates the capability of our method on crowd data ranging from low to high density.

The automatic motion pattern segmentation in a crowded environment has been an active topic in computer vision for more than one decade. Comparison of the methods is typically done on crowd videos downloaded from the web. The evaluation is done visually on selected videos. Authors often select a set of

examples where they have better results than the competing method. Some works annotates the boundary of the crowd and its main direction, but the evaluation only compares the correct direction of the motion patterns. To our knowledge only one public crowd database provides annotation and evaluation respecting motion patterns. The CUHK database [4] has annotation for 300 videos. In each video, a set of 30 frames was selected and detected tracklets were grouped based on the collective motion of individuals. The representation is similar to motion pattern definition, but instead of region only detected tracklets are used to define group segregation.

Moreover, we have found inconsistency and many mistakes in data annotation of the CUHK dataset; see Sect. 3 for more details. The re-annotation we made for the subset of the CUHK database draws the boundary of the motion patterns for a set of frames and captures motion overlaps among motion patterns. The proposed annotation helps to reveal the groupings consistency across a given set of frames. We have benchmarked our results against two state-of-the-art methods and show that in the situation where groups are well segregated (low-motion overlap across set of frames) all state-of-the-art methods provide reliable output. The performance deteriorates with increasing level of mixed motion in the input data.

1.1 Related Work

Segmentation of crowd scenes based on motion has attracted a significant amount of research works. Obtained motion patterns can be used in a wide range of applications like tracking in crowd [7], sink and source seeking [8], or anomaly detection [5]. A recent survey on motion pattern segmentation methods can be found in [1]. Here we concentrate on methods missing from the survey.

The feature extraction is a crucial step and each subsequent task benefits from its clear representation. A high density of people in the crowd performing various irregular motion leads to frequent occlusions. Traditional object detection and tracking algorithms, which are also computationally expensive for a large number of objects, often fail in the case of severe occlusions [1]. The trend is to use pixel-based features, because in high density crowds the local motion features such as point trajectories can be obtained more easily. The generalized KLT tracker [4,6] or Lagrangian framework [5] are quite popular. The provided trajectories capture the actual motion of the crowd and are useful for wide fields of view with low resolution.

Promising results have been shown in [5] where coherent motion is detected based on an analysis of Lagrangian particles. On top of that two-step clustering is applied to construct stable semantic regions from detected time-varying coherent motion. They made a comparison to six state-of-the-art methods on a selected subset of videos. Neither data with annotation or code is provided to the public. Thus we are unable to compare our work to such a promising method.

An interesting characteristic called the crowd collectiveness descriptor has been proposed in [3]. The collectiveness indicates the degree of individual acting as union in collective motion. The descriptor measures the collectiveness in single

frame. The grouping consistency between frames is not maintained at all as shown in Fig. 1. Nevertheless the method has a great potential to be extended into temporal domain as shown in [6] and [4].

Collective density clustering was proposed in [6] to recognize local and global coherent motion for varying crowd density. The method was benchmarked against [3] on the Collective Motion Database. The goal was to compare the correct level of collective motion (low, medium, and high). There are only visual examples showing the motion pattern segmentation against state-of-the-art method and one comparison on correct direction detection. The code is not provided therefor we can not compare our work to this method.

In [4] the crowd collectiveness descriptor is used to learn the collective transition priors from a given set of frames. The crowd is analyzed at the group-level where a group is considered as a set of members with a common goal and collective behaviors. The approach provides consistent group segregation in a crowded environment with low motion overlap. Yet the method fails to maintain the consistent groups segregation in complex crowded environment as illustrated in Fig. 1. The groups segregation is equivalent to the motion pattern definition we have. The authors provide a database with annotation and source code. The comparison can be found in our experiments.

We conclude our review with a recent method for object segmentation driven by motion [2]. The graph among each point of a trajectory is formed and minimum cost multicuts function is applied to precisely segment moving object from the scene. The method has incorporated motion, color and spatial distance into the graph to represent the relation between trajectories. The grouping consistency is maintained, but color distance can assign similar individuals with different motion to same group. On top of that it requires extensive training with precisely annotated training data. As the source code is provided we compare our work to this method.

The main contributions of our approach can be concluded as follows:

- We show how to simply extend the crowd collectives descriptor designed for a single frame to the temporal domain by averaging the point trajectories across fixed set of frames. The proposed framework results in temporally consistent groupings of individuals capable of segmenting arbitrary shapes of groups for structured and unstructured crowds with various crowd densities.
- Temporal analysis of point trajectories across a set of frames introduces the problem of overlapping motion patterns. Our framework does not require any training, only one parameter is hardwired. The number of neighbours for k-NN graph construction is set to represent up to four overlapping motion patterns in a particular image area.
- The threshold for graph segmentation is not fixed but it is data-driven and thus better represents the actual motion presented in the crowd. The results on subset of the CUHK dataset show higher accuracy for overlapping motion patterns against the state-of-the-art method.

2 Motion Pattern Segmentation Framework

The idea is that collective motion in the crowd can be discovered by temporal analysis of points trajectories. The analysis of long-term point trajectories has been very popular recently. It played an important role in works related to motion segmentation [2,4,6]. These trajectories are called tracklets, and provide pixel movement in a given set of frames. Nevertheless other optical-flow-based trajectories like large displacement optical flow [9] can be used as input to our framework. Here we stick to dense trajectories presented in [10] since they can filter out movement caused by camera motion. The basic setting is used providing reliable points tracklets for 15 consecutive frames.

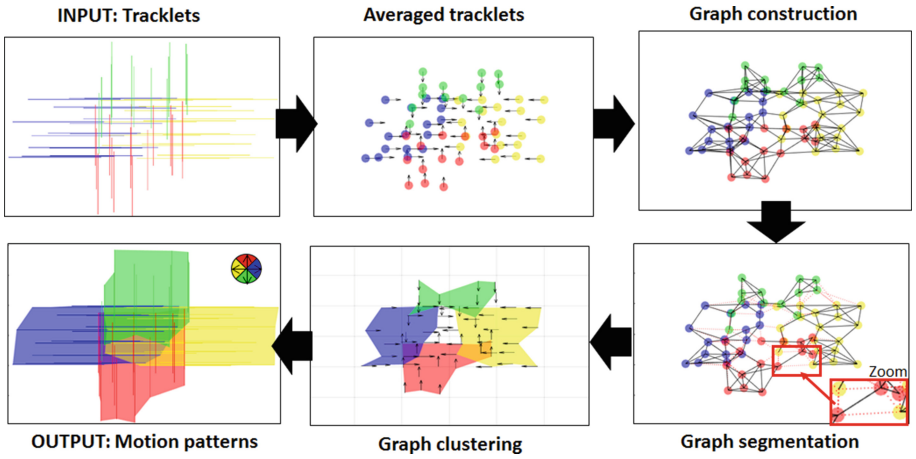

Fig. 2. The framework simplified flowchart. The graph is constructed so that each node (averaged tracklet) is connected by edges to k-NN nodes. During the graph segmentation the edges with low collectiveness are cut-off (red dotted lines). Finally, clustering is applied and connected nodes with similar motion are merged. Best viewed in color. (Color figure online)

We use the similarity graph from the spectral clustering [11] to represent the topological structure of tracklets detected in crowded video scene. We show how to take advantage of the collectiveness descriptor [3] and design criteria for the automatic threshold selection used for the graph segmentation. Figure 2 shows a basic simplified flowchart. The input shows trajectories of four points groups with different directions. Each group of points is starting at different sides of the image and the overlap of their trajectories occurs in the middle of the image. The tracklet of each point is averaged for a given period of time (in this case whole tracklet) and the graph is constructed based on the k-nearest neighbor (k-NN) of averaged tracklets. During graph segmentation with the automated threshold selection, edges with dissimilar motion are cut-off. Only edges with high collective

motion remain in the graph. Having obtained a graph connecting nodes with high collective motion, clustering is applied to find coherent motion patterns. The motion patterns can be used in various applications. In this particular work, we concentrate on the overlapping motion pattern segmentation and its evaluation on the CUHK dataset [4]. The individual components of the framework are described next.

2.1 Graph Construction

Interactions of neighbours play an important role in crowded environments. To capture the topological structure of these interactions we construct a similarity graph such that every averaged tracklet is represented by a node v

$$v = (\bar{x}, \bar{y}, \partial x, \partial y) \tag{1}$$

where the first two values are coordinates of the averaged tracklet and the last two represent the average velocity of the tracklet. Every node v is connected via a edge to each of its k-nearest neighbours nodes. Adjacency Matrix A is used to represent the oriented k-NN graph. Matrix rows and columns are labeled with graph nodes $v \in V$. Matrix entries at position (v_i, v_j) are 1 or 0, depending on whether v_i and v_j are connected or not. The matrix A has zero entries on the diagonal, since no tracklet connects to itself.

First we create an adjacency matrix based on k-nearest neighbours of each node. Each line of matrix A (each node with its neighbours) forms a small group of tracklets. In a real scenario noise is present in the data and single distant nodes can connect to the graph. Thus a distance threshold is applied to remove distant nodes from the matrix A with entry 1 as follows

$$A_{Dist}(v_i, v_j) = \begin{cases} 1, & \text{if } A(v_i, v_j) = 1 \ \wedge d(v_i, v_j) < \mu_D + \sigma_D \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$\mu_d = \frac{1}{N} \sum_{A(v_i, v_j)=1)} d(v_i, v_j) \tag{3}$$

$$\sigma_d = \sqrt{\sum_{A(v_i, v_j)=1)} \frac{1}{N} (d(v_i, v_j) - \mu_d)^2} \tag{4}$$

where $d(v_i, v_j)$ is the Euclidean distance between two nodes, μ_D and σ_D is the average distance and the distance standard deviation between all nodes in matrix A, N is the sum of entries when matrix $A(v_i, v_j) = 1$.

Figure 3 illustrates the effect of k-NN number selection on graph clustering. In the case when a small number of neighbours is selected the collective motion can not be revealed (first two columns in Fig. 3). Thus we seek the minimum number of neighbors to get precise motion patterns. This number depends on the motion overlap present in the data. For no overlapping movement, $K = 3$ is enough, for partial overlap $K = 5$ is required. For complete overlap $K = 20$

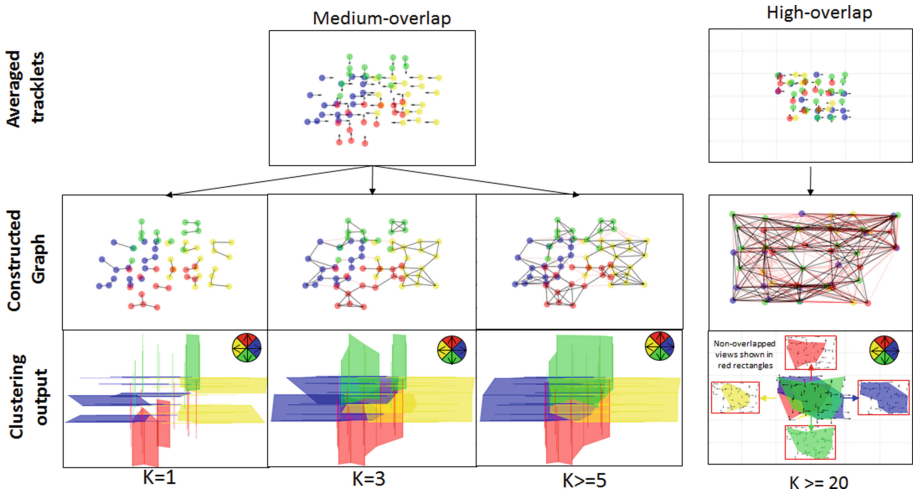

Fig. 3. Visual examples of graph construction for various number of neighbors. First example shows the graph construction for various numbers of k-NN *(first column $K = 1$ s $K = 3$, third $K = 5$)*. The second example shows the constructed graph for high motion overlap. For mixed up motion the minimum number of neighbours is 20.

is required. The lower value of k would result in scattered areas, while a higher number of neighbours would result in the same clustering output. The optimal number of k to reveal up to four mixed motion at a particular segment of the image in 15 consecutive frames is set to $k = 20$ based on empirical evidence. The larger number of neighbours also helps to overcome the noise produced by dense trajectories at a particular area of the image.

2.2 Graph Segmentation

The non-zero entries of adjacency matrix A_{Dist} are weighted based on velocity correlation among a given pair of nodes and forms weighted adjacency matrix

$$W(v_i, v_j) = \begin{cases} corr(v_i, v_j), \text{if } A_{Dist}(v_i, v_j) = 1 \\ 0, \text{otherwise} \end{cases} \tag{5}$$

The matrix W characterizes the behavior consistency among neighbouring individuals in the crowd. The maximum value of matrix W is 1 and it means that nodes have the same direction of velocity vectors. If two connected tracklets move in the same direction and have highly correlated motion we want to preserve the edge, otherwise the edge should be removed from the graph. By applying the threshold T on the matrix W, we ensure that only highly correlated neighbors remain in the graph. The selection of threshold from interval $T \in (0, 1)$ is driven by data. We make use of the collectiveness descriptor from

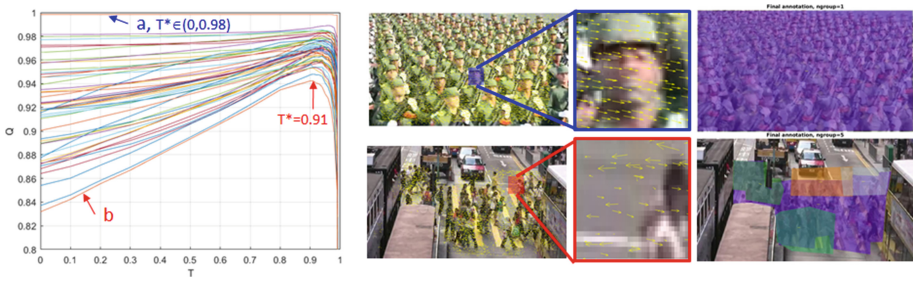

Fig. 4. The values of collectiveness criteria defined in Eq. 7 can be seen on the right side. On the left side examples for the highest and lowest criteria curve are shown. The top row shows a visual example for curve marked with a when the criteria value is close to 1 for the threshold interval (0,0.98). It is almost an ideal case where the velocity vectors have a similar direction, their correlation variation is not bigger than 2 %. The bottom row shows example for the curve marked with b when the criteria has the lowest value. It shows the hardest example of the CUHK subset. The detail shows that velocity vectors for both directions are mixed. The last column of the figure shows ground truth annotation for selected examples. It can be seen that the criteria value captures the level of motion overlap in input data.

[3] and define individual collectiveness for each node v_i and its neighbours as

$$\phi_{v_i}(T) = \frac{1}{N_i} \sum_{W>T} w(v_i, v_j) \tag{6}$$

where N_i is number of edges connected to node v_i after applying threshold T and ϕ_{v_i} thus represents the mean correlation value between node v_i and its neighbours v_j higher than threshold T. Each node and his neighbours form a small group. While in Zhou et al. [3] the collectiveness was computed for one fixed threshold, here we change the threshold to evaluate the collectiveness of each group. The $\phi_{v_i}(T)$ measures the individual collectiveness of the group. Now we can define collectiveness criteria for optimal threshold selection as follows

$$T^* = \underset{T \in <0,1>}{argmax} \left(\frac{N_T}{N_G} \cdot \frac{1}{N_G} \sum_{W>T} \phi_{v_i} \right) \tag{7}$$

The criteria measures normalized average collectiveness for all individual groups, N_G is the number of groups, and N_T is the number of groups with individual collectiveness bigger than the threshold. Figure 4 shows the value of the criteria for threshold values in interval $T \in (0, 1)$ on a subset of the CUHK dataset. For each video segment we have one criteria curve. The value of the criteria for the minimum threshold $T = 0$ serves as a measure of how much structured motion is present in the video segment. The criteria value for each threshold tell us how big the collectiveness is across all groups formed in the graph.

In the ideal case (top line in Fig. 4) when velocity vectors are highly correlated across the whole sequence, the criteria curve remains the same for a long interval

of threshold T. This tell us that graph nodes are well separated no matter which value of T we select and the motion of each group is highly correlated.

The decrease in criteria value is caused by lower correlation between the node and its neighbours which form a group. The threshold value $T = 1$ means that we don't allow any velocity difference between connected nodes. Lowering the threshold means that we allow some difference, but the question is how to define the exact of the velocity variation that is permitted? We seek the optimal balance between the number of groups and their average collectiveness represented by the maximum value of the criteria defined in Eq. 7. The threshold for each video sequence is optimized using the golden section search method. After applying the threshold T^* on matrix W each node preserves the edges with high correlation and cut-off the rest.

2.3 Graph Clustering

Having obtained a segmented graph that forms small compact groups (neighbours around each node) we can proceed with clustering. The existing graph edges maximize average collectiveness across the whole video segment. But the segmented graph tell us only the link between closest nodes. Now we want to cluster the graph nodes to form the motion pattern.

Figure 5 illustrates the clustering on a simple example. The decision whether to merge the neighbour's nodes depends on the correlation between the parent node and its children. The threshold value for graph segmentation defines how big variation is allowed between two nodes. Ideally this value would be set to 1 and only same direction nodes will be connected. In real data the threshold is typically around 0.95 which means 5 % variation between two nodes is allowed. We recursively search through the graph and look if parent/children correlation fits the threshold. If all the children meet the threshold the neighbour node is merged in the same cluster as the parent node. If any of the children violate the correlation threshold the recursive search for the neighbour node is stopped.

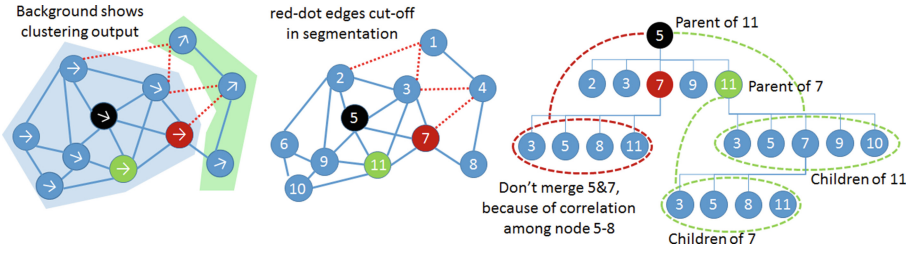

Fig. 5. Visual example of graph clustering through recursive search. Here we start from node 5 and concentrate on neighbour nodes 7 and 11. If the correlation between all of the children nodes and the parent node fits the threshold T the neighbour node 11 is assigned to the same cluster as node 5. Otherwise the recursive search is stopped. Node 7 is eventually assigned to the same cluster through node 11.

The clustering reflects arbitrary shapes appearing in the crowd while allowing varying size of clusters. The proposed framework is capable of segmenting motion patterns for various levels of density, from highly aggregated motion to low density disconnected areas.

3 Experiments on the CUHK Dataset

The proposed framework is evaluated on a challenging subset of the CUHK video dataset [4]. We first describe the details about the dataset and drawbacks of the provided annotation. Then we demonstrate the results obtained for our method versus state-of-the-art [2,4].

The CUHK dataset provides annotation for 300 segments for varying length of video segment ranging from 7 up to 67 frames. Annotation of the data was made for tracklets found by the generalized KLT tracker. The tracklets are grouped based on the criterion that members in the same group have a common goal and form collective movement. Tracklets not belonging to any group are annotated as outliers. Figure 6 show inconsistency and mistakes observed in data annotation. More than 50 % of data have inconsistency in terms of merging distant groups versus splitting neighbours to different groups. Moreover serious mistakes like ignoring the opposite direction of individuals or noise tracklets assigned to the nearest group were discovered. Thus we have decided to re-annotate the database.

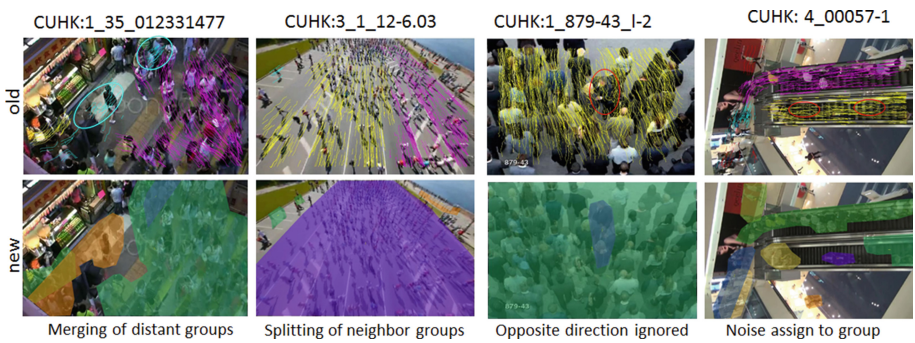

Fig. 6. Figure illustrates the annotation inconsistency (first two images in top row) and errors (last two images in top row) in the CUHK database [4] and our re-annotation (bottom row). First image shows merge of the individuals in blue circles who are separated by individual moving in different direction. In contrast let's look at the second image with pedestrians running in a marathon. All pedestrians are running in same direction, there is no obvious reason to separate the groups, thus we keep them as one. The third image shows that the opposite direction was ignored and merged with the main direction. In the fourth image the escalator movement is merged with the main direction (red circles show observed errors). (Color figure online)

The full CUHK dataset contains repeating scenes clipped in separate video segments. From the set of 300 annotated video segments there are actually 140 unique scenes, the rest are the same views for different time window. The database consist of scenes like marathon runners, military marching, protesting pedestrians, escalators, cross walk, public transport stations, shopping malls or people walking in the street.

We have further divided the database to simple and complex scenarios. In simple scenarios the groups of individuals can be easily segregated while in complex scenarios the individuals are mixed in a given set of frames. Mixed individuals means that at a particular part of the image pedestrians move in more than one direction (e.g. cross-walk). Both scenarios typically have a crowd moving in one or two main directions with some individuals moving independently. Two-thirds of the CUHK data contain simple scenarios, the rest is considered as complex scenarios. Some of the complex scenarios are so challenging that annotation of such videos would be very arguable. Thus we selected 25 out of 40 complex scenarios and 25 out of 100 for simple scenarios.

The selected 50 videos spans a wide range of crowded scenes covering different pedestrian size and various crowd densities. Figure 6 compares the original annotation to our new one. The ground truth provides boundary information and main direction of the motion pattern. Two annotators cross-validate their outputs. The boundary is drawn to capture spatial region with coherent flow in a given set of frames. The various speeds (e.g. individuals running on escalators) are ignored. The same starting frame as in [4] was selected for data re-annotation, each re-annotated set last 30 frames. In total 1500 re-annotated frames are used as a benchmark. Our annotation of groups boundaries for a given set of frames helps to reveal the consistency of individuals grouping between frames.

3.1 Results on CUHK Data

In order to evaluate the framework we compare its performance against recent state-of-the-art methods proposed in [4] and [2]. We run available binaries on the subset of the CUHK data comprising 1500 re-annotated images. The obtained results are treated as a clustering problem. To benchmark the performance of the methods we use the Purity and F-measure.

The density of point trajectories between our method and state-of-the-art differs, see Fig. 7. Therefore the computation of the true positives (TP), false positive (FP) and false negatives (FN) is based on overlap between the ground truth area and the area drawn around clustered point trajectories.

The cluster switching between frames is misleading. Building an application for abnormal activity recognition on top of the method with cluster switching between frames would produce an enormous number of false alarms. By using the area overlap and evaluation across the whole sequence of 30 frames we also penalize the clustering inconsistency when clusters switch between frames.

The state-of-the-art method has various parameters that can be change for boosting the performance. For the method in [4] there are 8 parameters. As the available code was tuned for the CUHK dataset we have kept the original setting.

Table 1. Results on the subset of the CUHK database for simple (left) and complex (right) scenarios. We report results for **D**: purity, **P**: average precision, **R**: average recall, **F**: F1-measure. Subscript W represents clusters weighted by their size. The first three rows show unweighted clustering, the last three rows show clustering weighted by the size of the cluster

	Simple scenario				Complex scenario			
	D	P	R	F	D	P	R	F
Our	**81.70 %**	**78.88 %**	**85.92 %**	**81.96 %**	**70.51 %**	**72.16 %**	**81.81 %**	**75.77 %**
Shao et al. [4]	41.37 %	52.77 %	46.06 %	47.43 %	38.04 %	44.68 %	46.70 %	45.08 %
Brox et al. [2]	58.47 %	63.54 %	52.58 %	55.85 %	47.52 %	58.72 %	43.16 %	47.74 %
	D_W	P_W	R_W	F_W	D_W	P_W	R_W	F_W
Our	**91.62 %**	**90.42 %**	**95.26 %**	**92.47 %**	**84.00 %**	**80.55 %**	**91.10 %**	**84.74 %**
Shao et al. [4]	68.10 %	81.28 %	69.16 %	72.27 %	65.11 %	71.86 %	72.52 %	71.27 %
Brox et al. [2]	81.45 %	76.25 %	72.33 %	71.17 %	74.13 %	68.31 %	61.63 %	62.00 %

For the method in [2], we use the available binaries and run the code against the crowd data to see how well the parameters tuned for object segmentation work for crowded environments. Additionally there are two independent parameters: number of frames and threshold. We observe that changing the number of frames from 8 to 15 did not significantly affect the results (change of F-measure was around 0.01 %). Only the threshold influences the final number of clusters in the video. The best results were obtained for threshold 0.5.

Table 1 provides two types of results, unweighted and weighted. For unweighted results the F-measure uses average precision and recall across all the clusters in the given video. For weighted results the precision and recall for each cluster is weighted by the proportional cluster size. For the unweighted F-measure we outperform the state-of-the-art by 30 %. Such a big difference in performance is caused by missed irregular clusters. For instance a single individual walking in a different direction than main clusters, see example in Fig. 7. For the weighted F-measure the performance of state-of-the-art increases by 25 %, since the methods capture the main clusters. Nevertheless we are still 20 % ahead in performance, since our method does not suffer from cluster switching between frames. The methods in [2,4] provide good results for main clusters but lack the capability of detecting the irregular motion of individuals. Moreover, distant individuals are clustered together even if they move in the opposite direction and method in [4] suffers from cluster switching between the frames.

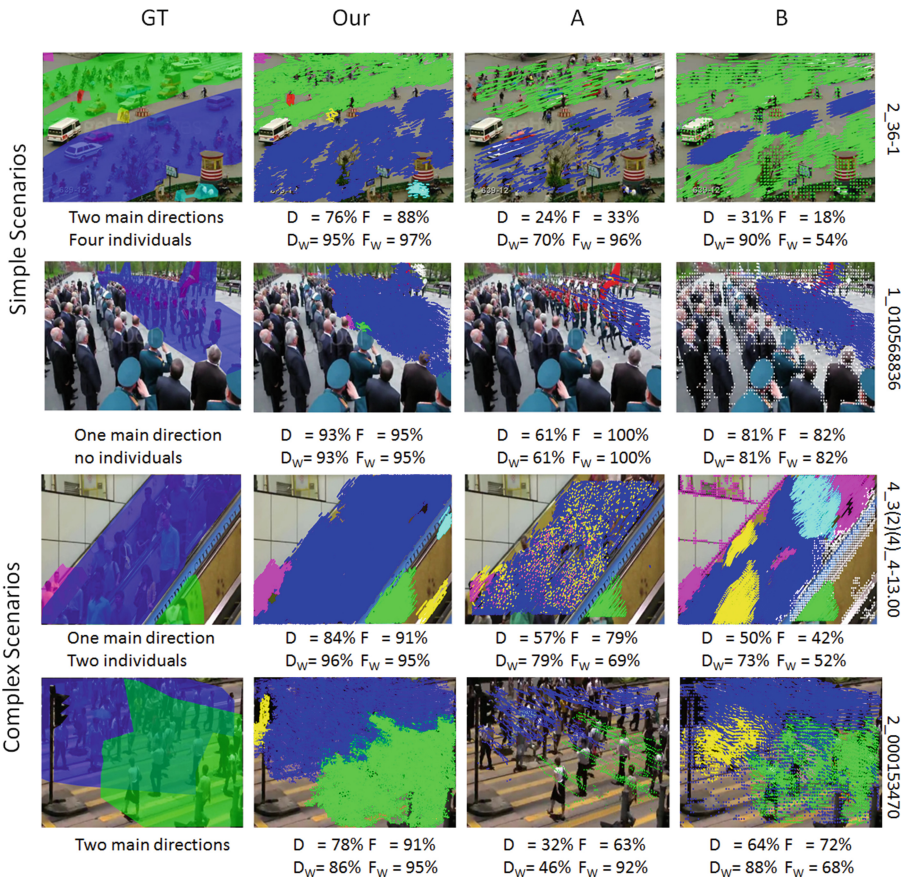

Fig. 7. Results for a simple scenario with easily separable groups and a complex scenario with mixed groups. The first column shows the ground truth, the next three columns show the output of our method, method A [4] and method B [2]. The D, F, D_W, and F_W are defined in Table 1. The first row shows a scene with two main opposite directions and several individuals highlighted by pink, red, yellow and cyan colors. Our method is capable of recognizing all motion patterns, method A can only recognize two main directions and method B incorrectly assigns two opposite direction as one and creates a separate cluster for cars with a different color from pedestrians. The second row shows a scene where a military unit is marching. All methods correctly cluster the scene. The third row shows a scene with escalators. Method A has inconsistent grouping. Pedestrians in the main direction highlighted in blue are assigned to the pink and yellow clusters for some of the frames. Method B merges distant pedestrians to one group highlighted in pink and over-segments the main direction into three clusters highlighted in blue, yellow and cyan. The last row shows a challenging cross walk scene where people mix together. Our method and method A correctly recognize the clusters. Method B over-segments the pedestrians moving down into two clusters and some distant point trajectories in the bottom are merged with the blue cluster although they move in opposite directions. (Color figure online)

4 Conclusion

This paper introduced a framework for motion pattern segmentation in crowded environments. The proposed method is fully unsupervised and uses short tracklets detected by dense trajectories. It reveals the collective motion of individuals independent of the crowd density. Our method can detect different scales of groups with arbitrary shapes and distinguish the big groups and irregular motion of individuals that move otherwise. The resulting grouping of individuals is temporally consistent over the set of frames, a property that requires post-processing in the existing approaches.

We have tested our approach on a subset of the CUHK database. Experimental results show that our approach outperforms existing state-of-the-art method by more than 20 %. However, we note that the annotation of motion patterns is subjective and differs between people. In future work we will evaluate our framework on practical applications like unusual event recognition and tracking in crowded environments.

We will conclude this paper by summarising the framework shortcomings. The average tracklet is computed across 15 frames. This was chosen based on the dense trajectory output and provides satisfying results for the given set of videos. In future this property should be also data driven as the dynamic of the crowd evolves in time. For rapidly moving objects, it might happen that the point trajectories are short. In these cases, we would increase the number of frames for computation of the averaged tracklet.

From our understanding of the framework the graph clustering part is the most weak and offers a real scope for improvement. Clustering of the graph should be done from a bigger perspective than just considering the parent and children of the neighbour node. We plan to apply a multi-resolution approach in order to reduce the over-segmentation of data.

Acknowledgement. This work was partially supported by the People Programme (Marie Curie Actions) of the European Unions Seventh Framework Programme FP7/2007-2013/ under REA grant agreement n[324359].

References

1. Li, T., Chang, H., Wang, M., Ni, B., Hong, R., Yan, S.: Crowded scene analysis: a survey. IEEE Trans. Circ. Syst. Video Technol. **25**(3), 367–386 (2015)
2. Keuper, M., Andres, B., Brox, T.: Motion trajectory segmentation via minimum cost multicuts. In: The IEEE International Conference on Computer Vision (ICCV), December 2015
3. Zhou, B., Tang, X., Wang, X.: Measuring crowd collectiveness. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3049–3056 (2013)
4. Shao, J., Loy, C.C., Wang, X.: Learning scene-independent group descriptors for crowd understanding. In: TCSVT (2016)

5. Wang, W., Lin, W., Chen, Y., Wu, J., Wang, J., Sheng, B.: Finding coherent motions and semantic regions in crowd scenes: a diffusion and clustering approach. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part I. LNCS, vol. 8689, pp. 756–771. Springer, Heidelberg (2014)
6. Wu, Y., Ye, Y., Zhao, C.: Coherent motion detection with collective density clustering. In: Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, pp. 361–370. ACM (2015)
7. Rodriguez, M., Sivic, J., Laptev, I., Audibert, J.Y.: Data-driven crowd analysis in videos. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 1235–1242. IEEE (2011)
8. Jodoin, P.M., Benezeth, Y., Wang, Y.: Meta-tracking for video scene understanding. In: 2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. IEEE (2013)
9. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Deepflow: large displacement optical flow with deep matching. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1385–1392 (2013)
10. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: IEEE International Conference on Computer Vision, Sydney, Australia (2013)
11. von Luxburg, U.: A tutorial on spectral clustering. Stat. Comput. **17**(4), 395–416 (2007)