Machine Vision and Applications (2023) 34:2
https://doi.org/10.1007/s00138-022-01348-0

ORIGINAL PAPER

Motion-region annotation for complex videos via label propagation across occluders

Muhammad Habib Mahmood[1,2] · Yago Diéz[3] · Arnau Oliver[2] · Joaquim Salvi[2] · Xavier Lladó[2]

Received: 27 January 2019 / Revised: 5 September 2022 / Accepted: 1 October 2022

Abstract
Motion cue is pivotal in moving object analysis, which is the root for motion segmentation and detection. These preprocessing tasks are building blocks for several applications such as recognition, matching and estimation. To devise a robust algorithm for motion analysis, it is imperative to have a comprehensive dataset to evaluate an algorithm's performance. The main limitation in making these kind of datasets is the creation of ground-truth annotation of motion, as each moving object might span over multiple frames with changes in size, illumination and angle of view. Besides the optical changes, the object can undergo occlusion by static or moving occluders. The challenge increases when the video is captured by a moving camera. In this paper, we tackle the task of providing ground-truth annotation on motion regions in videos captured from a moving camera. With minimal manual annotation of an object mask, we are able to propagate the label mask in all the frames. Object label correction based on static and moving occluder is also performed by applying occluder mask tracking for a given depth ordering. A motion annotation dataset is also proposed to evaluate algorithm performance. The results show that our cascaded-naive approach provides successful results. All the resources of the annotation tool are publicly available at http://dixie.udg.edu/anntool/.

Keywords Motion annotation · Motion segmentation · Tracking · Dataset

1 Introduction

Motion analysis is a prerequisite in video analysis with its applications in computer vision ranging from surveillance [1–3], multi-object tracking and crowd estimation [4–9] to gesture recognition [10,11], video object segmentation [12–17], behavior analysis [18,19] and anomaly detection [20–22]. An objective analysis of moving objects can be carried out when motion is accurately detected and segmented as a prior. In the state-of-the-art of computer vision, precise and robust algorithms, which can work in the presence of occluders and other distortions, while the acquisition of video is done from a moving camera, are still elusive. Therefore, research to find a solution of these tasks is still an open field.

To find a solution for each of these tasks, algorithms are needed to be devised, which are trained and tested on their corresponding annotated datasets for performance analysis. A prevalent problem while creating such algorithms is the availability of relevant utilizable corresponding datasets. The utility of a dataset is determined by the domain it is captured for, the variability it encapsulates and the comprehensiveness of the annotation it contains. While the acquisition of data and variability are addressable, a comprehensive annotation with respect to each task is cumbersome. In crowd estimation [4,6,8,9], each individual moving person is annotated as a unique entity. To do so in images and videos with hundreds of people is quite time-consuming. In video object segmentation [12,13,15,17], the contour around each moving object has to be annotated. While in motion, the size, field of view, depth of field and projective geometry of the object also changes. Annotating such complex scenes requires a lot of time of the expert user in terms of annotation.

Specifically in motion segmentation, the limitations prevailing in annotated moving objects' datasets are restricting the development of effective motion analysis tools. The diversity and complexity of a real-life motion captured in a collection of video sequences determines how representative the dataset is of the actual problem. If the annotated

✉ Muhammad Habib Mahmood
mhabib82@hotmail.com

1 Air University, Islamabad, Pakistan

2 University of Girona, Girona, Spain

3 Yamagata University, Yamagata, Japan

Published online: 08 November 2022

Fig. 1 First, middle and last frame of a moving object while entering and leaving the field of view in a video shot. Top: The white car enters and leaves the frame without occlusion and distortion. Bottom: The blue truck enters and leaves the frame while undergoing complete occlusion, change in heading direction and illumination, a significant alteration in relative size and experiences perspective distortion (color figure online)

datasets encapsulate limited motion diversity, then the algorithms tested on them will also have limited applicability. On the other hand, if more complex motions are captured in a sequence for dataset formation, the dataset will become more representative but the task of correctly generating ground-truth motion label for each moving object in all the frames of a video sequence becomes increasingly cumbersome. Here, the problematic element is the expert-user annotation time, which increases as the captured motion becomes excessively complex.

An illustrative example is presented in Fig. 1, which shows the first, middle and last frames of two moving objects in a video shot, while they enter and leave the field of view. In the left top view of Fig. 1, the white car enters the scene, it continues its movement from left to right in the scene as shown in the middle top image of Fig. 1 and the same white car moves toward the right end of the field of view to exit the scene as shown in the right top image of the Fig. 1, whereas, in the left bottom view of the Fig. 1, the blue truck enters the scene, it continues its movement from right to left in the scene as shown in the middle bottom image of Fig. 1 and the same blue truck after taking a turn moves toward the right end of the field of view to exit the scene as shown in the right bottom image of Fig. 1. The white car in the top row remains unoccluded, relative change in size across all frames is minimal, the illumination remains generally homogeneous, and no perspective distortion effect can be seen. On the other hand, the blue truck, present in the bottom row, enters the field of view with a small size due to being considerably deep in the scene with reference to the camera, experiences complete occlusion during the course of its motion and exits the frame with an enlarged size, change in heading direction, variation in illumination and with perspective distortion. The expert-user annotation time for generating ground-truth on these two motion samples is radically different. While the annotation labels on the white car be provided with state-of-the-art label propagation algorithms, there is no modern, time efficient methodology or platform, to annotate the blue truck or such motions.

This limitation in label propagation can be looked into as a set of multiple subproblems based on the complexity and variation in the object motion. The variants include a considerable change in size or illumination, partial or complete occlusion, static or moving occluder, multiple appearance and disappearance in the field of view (FoV), perspective distortion, etc. Each variant, if tackled separately, with a unique approach, can yield improved results.

In this paper, we propose a methodology, which utilizes the expert-user time to propagate labels on all moving objects in all the frames of a video sequence captured from a moving camera. With an existing platform [23], which propagates labels in situations with no occlusion nor distortions, our methodology is integrated to propagate labels across occlusions and its related distortions. The propagation result keeps

the object shape intact with scale adjustment. We do so by using just two user-labeled motion masks, the first and last frame of a subproblem set. Utilizing the two masks, we perform object mask propagation across all frames using maximal flow vector count, acquired through Large displacement optical flow (LDOF) [24]. Concurrently, we take a static occluder shape input on a single frame from the user, to perform occluder mask tracking using keypoint descriptors (SURF features [25]) across all frames. With non-rigid point set registration [26,27] of the first frame mask onto the last frame, we perform object mask scale adjustment to improve the propagated object mask estimate. To validate the performance of our approach, we carry out a quantitative and qualitative analysis of our algorithm on moving objects undergoing partial occlusion, where occluder is both static and moving, with sequences captured from a moving camera. In this regard, we used a 25 sequence occlusion/occluder dataset with moving objects going across static or moving occluder(s). On 20 static and 5 moving occluders, our results demonstrate that by splitting the motion annotation problem into subproblem sets, the expert-user time is utilized in an improved manner, maintaining accurate boundaries on the object annotations.

2 Related work

In general, the solutions of the video annotation problem try to achieve two distinct objectives, either to reduce the expert-user annotation time in generating the ground-truth of large-scale video data, or to improve annotation quality. These objectives are usually achieved by two approaches. One is to put forth comprehensive video annotation platform tool, which can label motions or objects of interest in video sequences as a standalone package. The other is to devise label propagation methodologies, which can be incorporated in the existing tools. The state-of-the-art in video annotation includes techniques from both practices.

Several video annotation tools have been developed in recent years. Predominantly, computer vision and machine learning methods are used as support for efficient human annotation. The different tools can be distinguished based on the functionalities they support. The pioneering work on video annotation was presented in *ViPER* [28], which was a reconfigurable video performance evaluation resource. It provided an interface for manual ground-truth generation, an evaluation metric and a visualization tool. It was a Java-based desktop application, which propagated rectangular or polygon region-of-interest (ROI) through linear interpolation. Similar desktop-based *GTTOOL* [29] and web-based *GTTOOL-W* [30] tools were presented, with a goal to improve user experience with respect to *ViPER* [28] by providing edit shortcuts, and by integrating some basic computer vision algorithms to automate. The collaborative web-based implementation featured an easy and intuitive user interface that allowed instant sharing/integration of the generated ground-truths. The label propagation in these tools were performed using tracking approaches.

Relatively recently, a popular online, openly accessible tool *LableMe-Video* [31], was presented that allows annotation of object category, motion and activity information in real-world videos. This tool used homography to propagate the label across key frames in the video. With the same focus, *iVAT* [32], an interactive video annotation tool, which supports manual, semiautomatic and automatic annotations, was presented. This tool integrated several computer vision algorithms working in an interactive and incremental learning framework. Another human-in-loop methodology [23], to create ground-truth for videos containing both indoor and outdoor scenes, was used with the idea that human beings are experts at segmenting objects and inspecting the match between two frames. The approach contained an interactive computer vision system to allow a user to efficiently annotate motion. Similar tools for trajectory-based datasets have also been presented [33]. A comparative overview of the discussed annotation tools is given in Table. 1.

Besides independent annotation tools, some recent work has been presented solely related to label propagation, where a manually given object label in key frames is propagated forward and/or backward in all frames the object exists. Probabilistic graphical models for multi-modal label propagation in video sequences were used in [34–37]. An expectation maximization (EM) algorithm propagates the labels in a chunk of video with starting and ending frames already labeled. The unlabeled parts of the video are dealt within a batch setting. In [38], a similar approach was used to train a multi-class classifier. The pixel labels estimated by the trained classifier were fed into a Bayesian network for a definitive iteration of label inference. A hybrid of generative propagation and discriminative classification in a pseudo time-symmetric video model enables conservative occlusion handling. Moreover, in [39] the limitations of pure motion and appearance-based propagation methods were shown, especially the fact that their performances vary on different type of videos. To avoid these limitations, a probabilistic framework was proposed that estimated the reliability of the appearance-based or optical flow-based label sources and automatically adjusted the weights between them.

An active frame selection approach was adapted in [40, 41]. In [40], active frame selection was done by selecting k frames for manual labeling such that automatic pixel-level label propagation can proceed with minimal expected error. Here, the frame selection criterion is joined with the predicted errors of a flow-based random field propagation model. The method excels in utilizing human time for video labeling effectively. In contrast, an information-driven active frame,

Table 1 The comparison of existing video annotation tools

Comparison of existing video annotation tools												
Annotation tools	Properties of the annotation tools											
	Specifications				Features							
	Platform (D/W)	Prog.Lang. (J/CP/-)	Obj. Bdry. (R/E/P/A)	Object timeline	Key frames	Annot. Prop. (L/T/H)	Low depth variation	High depth variation	Occlusions (PR/C)	Multiple occlusions	Perspective variation	Multiple distortions
ViPER [28]	D	J	R/E/P	✓	✓	L	✓	X	X	X	X	X
GTTOOL [29]	D	–	P/A	X	X	T	✓	X	PR	X	X	X
GTTOOL-W [30]	W	CP	P/A	X	X	X	✓	X	X	X	X	X
LabelMe-V [31]	W	–	P	X	✓	H	✓	X	PR	X	X	X
iVAT [32]	D	CP	R/E/P	✓	✓	L	✓	X	PR	X	X	X

Their specifications and features. The table gives an insight about the features that the tool can handle automatically with less expert-user input. Acronyms are: D: Desktop-based, W: Web-based, J: Java, CP: C or C++, R: Rectangle, E: Ellipse, P: Polygon, A: Active Contour, L: Linear interpolation, T.: Tracking, H: Homography, PR: Partial, C: Complete. –: Not known, in Features X: This feature is not inherent. Expert-user time required

location and detector selection approach were used in [41]. The method optimizes on a given uncertainty bound, the selection of a detector at a particular location and also minimizes label uncertainty at each pixel. It also tries to optimize for computational cost for both manual and semiautomatic labeling. Other recent methods are [42].

More recently, a semi-supervised video annotation approach [43] was proposed by learning an optimal graph from a partially labeled object. The methodology also exploited multiple features, which could accurately embed the relationships among the data points. The similarity graph used the geometrical relationships among the training data points. The model was extended to address out-of-sample and noisy label issues. For an application of color label propagation, in [44], the problem of inferring color composition of the intrinsic reflectance of objects was addressed. The color labels are propagated between regions sharing the same reflectance, and the direction of propagation is promoted to be from regions under full illumination and normal view angles to abnormal regions. From another perspective, a diffusion approach for label propagation was used in [45]. The application of anisotropic diffusion on graphs and the corresponding label propagation algorithm on the vector bundles of Riemannian manifolds were presented. This definition of new diffusivity operators significantly improved semi-supervised learning performance.

The existing methodologies in label propagation address the problem in a limited range of applications. Though they perform well, they lack utility in real-life long videos in outdoor scenes, where multiple occlusions, stopping motion, perspective distortion, multiple appearance–disappearance and noise of camera motion are present. A reason for these limitations is the absence of a video dataset where these optical phenomenons could be tested. The recently presented UdG-MS datasets [46,47] contain these real noises, which makes their quantitative testing possible.

With our current proposal, we aim to tackle these prevalent shortcomings in the label propagation methodology. Results in the state-of-the-art demonstrate that the use of the semiautomatic, as well as the automatic, modality in annotation drastically reduces the expert-user time while preserving the quality of the annotations. We propose a semiautomatic approach by taking annotated labels on two key frames (first and last). We utilize LDOF to promulgate labels across occluders, so that moving object labels are retained even after occlusion. A further refinement of propagated label mask scale is performed by using a non-rigid point set registration method. In this way, we not only improve labels on occluded objects but also in objects undergoing perspective distortion. Furthermore, we provided a consolidated evaluation to establish the usage of our scheme in real-life scenes. Our methodology is generally applicable on objects undergoing partial occlusion by static occluders, although it may also

be applied on objects undergoing occlusion by other moving objects.

Our label propagation algorithm is introduced in Sect. 3. The experimental setup and evaluations are presented in Sect. 4. A course of action on how to improve results is also suggested in Sect. 5, before concluding in Sect. 6.

3 Motion-region annotation

Motion-region annotation means tagging all the motion regions with a unique label per motion in a sequence of frames by a human expert. More formally, given a sequence of N frames $f = \{f_1, f_2, \ldots, f_N\}$, the objective is to segment all the moving objects M with the set of labels $m = \{m^1, m^2, \ldots, m^M\}$.

As the goal of annotation is to generate the ground-truth for a given video, it is imperative to take the accuracy of the annotation into consideration. One way of maintaining accuracy is to generate annotation of one motion m^x at a time, with respect to their depth ordering in the scene. The object near the camera first (the one with least depth) and the object farthest from the camera last (the one with most depth). The depth order can be kept track of by the expert user. Hence, the objective is to find the annotation labels m^x where $x = \{1, 2, \ldots, M\}$, sorted by depth ordering, 1 being least deep and M being the deepest.

An underlying premise of all annotation tools is to utilize the expert-user time in an efficient manner. The tool presented in [23] facilitates the annotation of moving objects in a sequence of frames. An expert user defines the object outline contour in a key frame. The region inside the object contour is given a label, and then, the labeled contour is propagated both ways, forward and backward. The algorithm works well for moving object annotation, when the object does not undergo any occlusion or perspective distortion. In the presence of occlusion, perspective distortion and change in object's depth, the propagation fails. If the propagation fails due to illumination variance, background homogeneity with moving object, etc., the labeled region contour can be corrected in frames with bad annotations. The manual correction by the user is then linearly interpolated across all frames the label was propagated on. In the absence of real noise, the platform utilizes expert-user time efficiently and exhibits good results. On the contrary, it fails in real sequences, especially outdoors, where occlusion, change in depth and perspective distortions are somewhat dominant.

From another perspective, let us consider the sequence of frames shown in Fig. 2 as an example. The moving object enters and exits the FoV in the frames f_1 and f_{113}, respectively. The motion annotation of this object, m^1, in these 113 frames can be divided into a set of three subproblems. One from f_1 till f_{75} (m_1^1), when the object is fully visible without occlusion, which as mentioned earlier, can efficiently be handled by [23]. The second subproblem ranges from frames f_{76} till f_{100} (m_2^1), when the object is occluded by multiple static occluders, where this method [23] and other similar methods fail. The third subproblem is m_3^1, when the object is again fully visible from frames f_{101} till f_{113}, until it goes out of the FoV. Then, the overall motion-region annotation of the object, m^x is given by,

$$m^x = \bigcup_{i=1}^{S^x} m_i^x \tag{1}$$

The annotation task of each motion x, to be labeled in the sequence of frames, leads to its corresponding subproblem set S^x. Hence, the goal is to devise m^x, the moving object segmented mask over all frames in which the object is present, for all x. In the example given in Fig. 2, x, is the object label by depth ordering, and S^x, is the number of subproblems x^{th} motion-region annotation task was divided into. So, with x being 1 and S^x being 3, the labeled motion-region output of the framework for one object in the given example is given as,

$$m^1 = \bigcup_{i=1}^{3} m_i^1 \tag{2}$$

3.1 Motivation illustration

The question really is why should the annotation problem be considered as a set of multiple subproblems of annotation labeling. The goal of devising any annotation tool or labeling method is to reduce the expert-user annotation time. As illustrated in the example with reference to Fig. 2, the expert-user need not be engaged in the annotation of frames which the existing tool [23] can handle. As mentioned earlier, the tool [23] works well in the absence of distortions such as partial or complete occlusion, perspective distortion and change in object's depth and size. In these cases, as the tool [23] performs blind linear interpolation of the object mask, it fails to capture the nonlinear visual evolution of the object in the scene. Hence, it makes sense to divide the annotation problem of a moving object over all frames into two type of modular tasks, once where these distortions are absent and the other where these distortions are present. The expert-user gets engaged in the separation of these subproblems and only annotates the moving object for the subproblem with distortions.

A modular approach to solve this annotation problem can yield better results in terms of pixel accuracy and time efficiency. This approach of creating subproblem tasks facilitates the expert-user to objectively divide the annotation problem based on the behavior each moving object exhibits,

Fig. 2 Six frames of a moving object, black car, entering and leaving the field of view in a video shot. Top: The black car enters the field of view in f_1 (Left) and moves till f_{75} (Right), without occlusion 'm_1^1'. Middle: Here, the car undergoes partial occlusion by multiple static occluders from f_{76} till f_{100}, 'm_2^1'. Two frames in this subproblem, where the object was undergoing occlusion are shown, f_{82} (Left) and f_{90} (Right). In f_{82}, the object has started undergoing occlusion behind the two static occluders. In f_{90}, the object has almost gone across the occluders. Bottom: The car moves from f_{101} (Left) without occlusion till f_{113} (Right) when it completely goes out of the FoV,'m_3^1'

and also, inherently reduces user annotation time. This subcategorization based on label propagation complexity can further reduce the manual annotation time, if the label propagation in the problematic subsets, (the ones which require most user corrections due to real distortions), can be automatized.

As mentioned earlier, while [23] works well in unoccluded, low depth change and no perspective distortion motions, it fails otherwise. As a smart hybrid approach, the framework in [23] was used for the subproblems where these distortions were not present. To annotate the subproblems with distortions, we propose a semiautomatic annotation

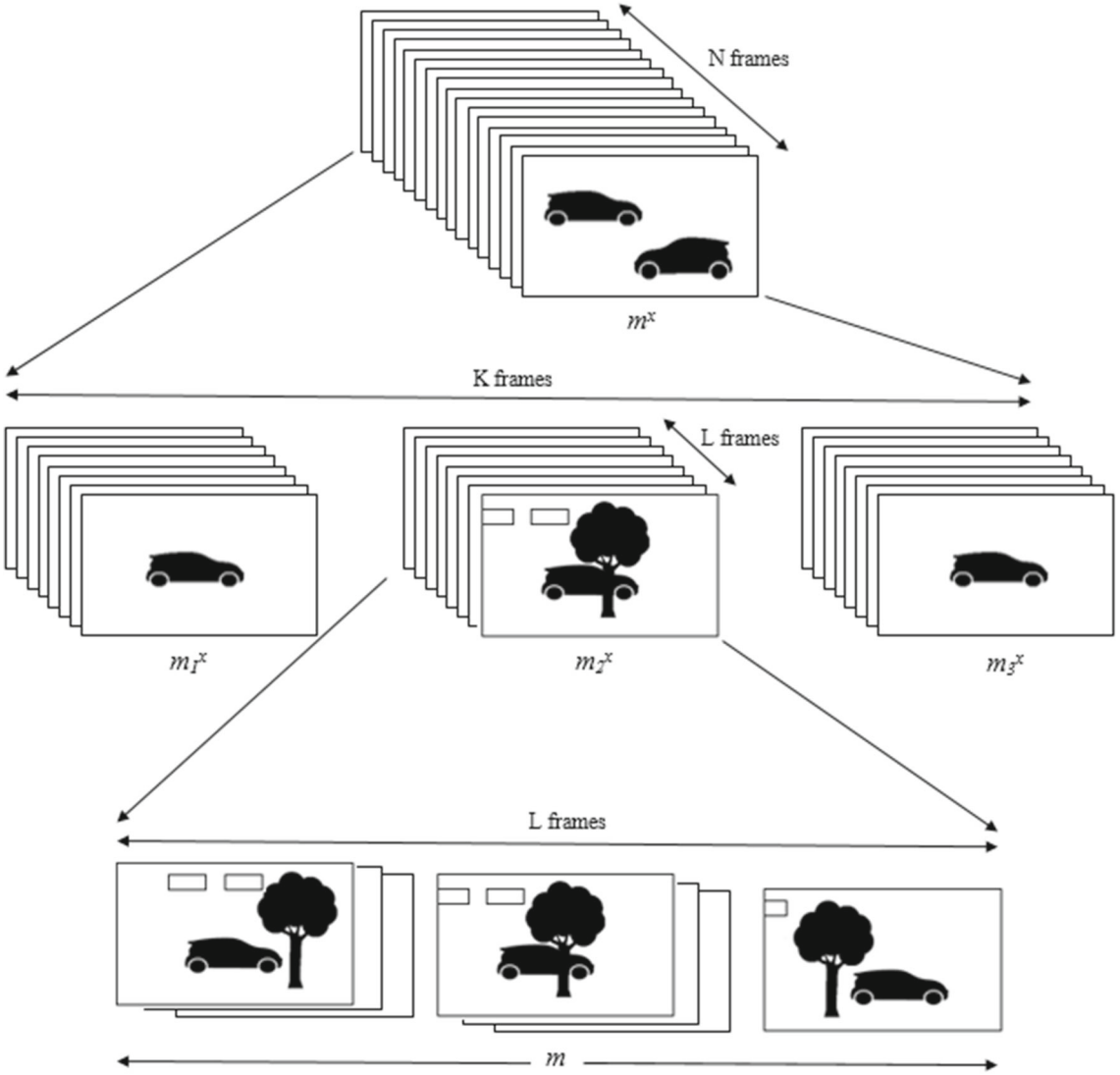

Fig. 3 Annotation flow of the motion-region annotation algorithm. The motion annotation subproblem of type-2 spanning over L frames is processed using the proposed algorithm

methodology to better utilize the expert-user time. In this section, our annotation algorithm is presented.

Given a set of K frames $f = \{f_1, f_2, \ldots, f_K\}$, with $K \subseteq N$, in which a single x^{th} moving object appears and then disappears from the FoV. The objective is to find the motion annotation label m^x. It is also given that the annotation task can be further divided into S^x sub problems, where each subproblem can have either of the two types;

- *Type-1 (motion under normal conditions)* Here, the object moves without occlusion or perspective distortion. The annotation under such moving conditions is computed through the work presented in [23].
- *Type-2 (motion under distorted conditions)* Here, the object undergoes occlusion and/or perspective distortion. The annotation under these conditions is resolved through our motion annotation algorithm.

A pictorial depiction of the same is given in Fig. 3. On the top, the figure shows a sequence of N frames, with two moving objects, so the objective is to estimate moving object labels $m^x = \{m^1, m^2\}$. Considering that object 1 is near the camera, it spans over K frames and the annotation task is divided into three subproblems $S^1 = 3$, then $m^1 = \{m_1^1, m_2^1, m_3^1\}$. Here, m_1^1 and m_3^1 are the subproblems of type-1, where the object does not undergo any occlusion or perspective distortion. This annotation problem is estimated by the framework in [23]. On the other hand, m_2^1 is the annotation subproblem of type-2, where the object experiences these distortions. If the movement under distortion spans over L frames, then the objective of the proposed algorithm is to

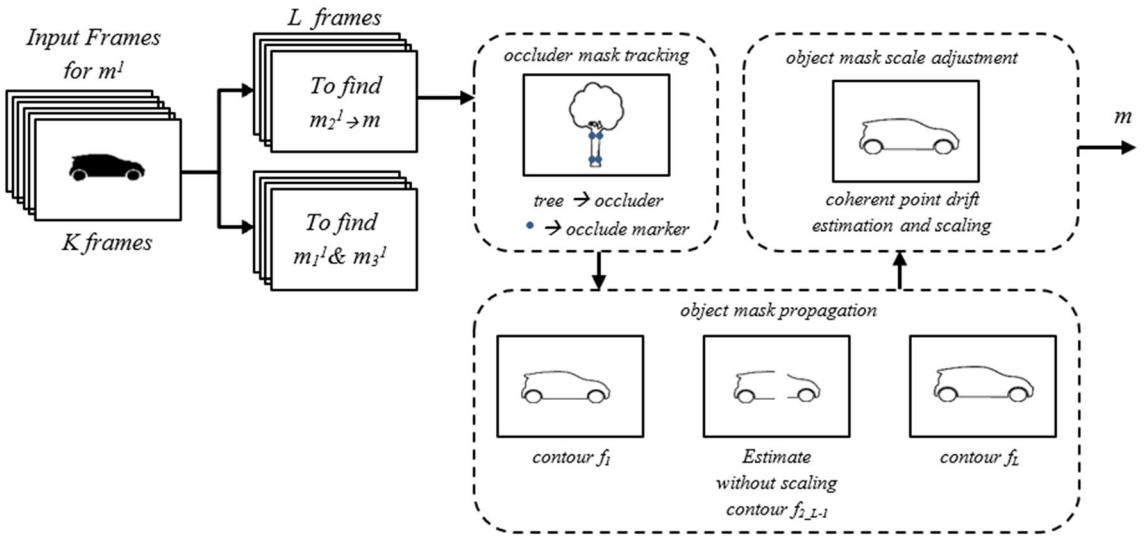

Fig. 4 Block diagram of the motion-region annotation algorithm of one moving object spanning over K frames. The motion annotation subproblem of type-2 to estimate m spanning over L frames is processed using the proposed algorithm

find m_3^1, given the expert-annotated object boundaries in the first and the last frame of the set L. A detailed account of the framework is explained further through Figs. 3 and 4.

In our work, any m_i^x is the output moving object label set computed for all the frames, in subproblem i of type-2, while annotating moving object x. For ease of notation, any such m_i^x in the remaining text is denoted as m. In our framework, a three pronged motion-region label propagation approach was taken to attain maximal accuracy with minimal expert-user intervention. The steps include *Occluder mask tracking* (m^{occ}), *Object mask propagation* (m^{ini}) and *Object mask scale adjustment* (m). A block diagram of the algorithm is shown in Fig. 4.

3.2 Occluder mask tracking (m^{occ})

In a subproblem with distortion, given a set of L frames and the occluder shape marker points, $\mathbf{P}_{f_1}^{occ}$ in the frame f_1 as inputs, the objective of *occluder mask tracking* was to perform shape tracking of the occluder mask in all the remaining $L-1$ frames. Here, the set of frames L in the subproblem is a subset of the total number of frames N, hence, $L \subseteq K$. The shape marker points of the occluder(s), $\mathbf{P}_{f_1}^{occ}$, in the f_1 frame of the set L were marked by the user through an interactive graphical user interface.

By taking the shape marker points, $\mathbf{P}_{f_1}^{occ}$, of the rigid occluder in the first frame as input, shape tracking of these markers was performed in the rest of the $L-1$ frames. With respect to this shape marker, the occluder mask in the first frame, $m_{f_1}^{occ}$, is given as

$$m_{f_1}^{occ} = region(countour(\mathbf{P}_{f_1}^{occ})) \tag{3}$$

while the complete occluder mask set is given as,

$$m^{occ} = \{m_{f_1}^{occ}, m_{f_2}^{occ}, \ldots, m_{f_{L-1}}^{occ}, m_{f_L}^{occ}\} \tag{4}$$

A user is required to define a set of markers (points) around the occluder such that they encapsulate the shape of the occluder. Subsequently, robust SURF features [25] inside the occluder mask, of this n^{th} frame, were estimated as, $F_n = SURF(m_{f_n}^{occ})$. After feature extraction, a point tracker was initialized on the user-defined occluder shape markers to estimate their probable position, in the following, $(n+1)^{th}$, frame. Given as,

$$\mathbf{P}_{f_n - f_{n+1}}^{occ} = PointTrackerEst(\mathbf{P}_{f_n}^{occ}) \tag{5}$$

The point tracker estimate in the $(n+1)^{th}$ frame was expanded on all sides by an expansion factor λ. The objective was to make sure that even in the case of wrongful tracking by the point tracker, the occluder must be inside the expanded mask. Surf features were again extracted in the λ-expanded mask.

$$F_{n+1} = SURF(region(countour(\lambda \mathbf{P}_{f_n - f_{n+1}}^{occ}))) \tag{6}$$

The features F_n and F_{n+1} were matched to yield feature pairs, which were then used to compute a similarity transform.

$$T_s = SimilarityTransform(FeatureMatching(F_n, F_{n+1})) \tag{7}$$

This similarity transform, T_s, multiplied with the input shape markers, $\mathbf{P}_{occ_{f_n}}$ results in the shape markers in the next frame.

$$\mathbf{P}_{f_{n+1}}^{occ} = T_s * \mathbf{P}_{f_n}^{occ} \tag{8}$$

Using eq. 3 for all n, the occluder mask for all the $L-1$ frames of a type-2 subproblem set with distortions, m^{occ}, can be estimated.

3.3 Object mask propagation (m^{ini})

Given the object mask in the first frame $m^{ini}_{f_1}$ and the last frame $m^{ini}_{f_L}$ of a subproblem set, the object mask propagation objective was to determine m^{ini}, where

$$m^{ini} = \{m^{ini}_{f_1}, m^{ini}_{f_1}, \ldots, m^{ini}_{f_{L-1}}, m^{ini}_{f_L}\} \tag{9}$$

The user-defined input masks are formed independent of occluder to save user time and effort. The output label set, which results when the first frame object mask is propagated forward till the last frame, is m^{ini}. This estimate can be utilized to perform nonlinear object scale adjustment in the subsequent step.

As a first step for label propagation, the forward optical flow, by using the state-of-the-art LDOF [24], was calculated. LDOF supports the estimation of dense optical flow field by integrating rich descriptors into the variational optical flow setting. In [24], the optical flow $\mathbf{w} := (u, v)^T$ is calculated with a comprehensive energy minimization term. These computed flow vectors give an estimate as to where each pixel moved in the following frame.

The given input, $m^{ini}_{f_1}$, contained the labeled pixels pertaining to the moving object region in the first frame. As for every frame n, the occluder mask $m^{ini}_{f_n}$ is known; then, for all n, $m^{ini}_{f_n}$ can be updated as,

$$m^{ini}_{f_n} = m^{ini}_{f_n} - (m^{ini}_{f_n} \cap m^{occ}_{f_n}) \tag{10}$$

For f_1, it becomes $m^{ini}_{f_1} = m^{ini}_{f_1} - (m^{ini}_{f_1} \cap m^{occ}_{f_1})$. Following this occluder mask subtraction update in the object mask, a set of forward flow vectors of all the pixels in $m^{ini}_{f_1}$ were segregated. In effect, this set contained the pixel-movement estimated by LDOF for all the pixels in the object region. It can be seen from Fig. 5 that though the vector directions are robustly detected inside the homogeneous region of the moving object, the estimates around the object boundary are adrift. Hence, as an initial estimate, instead of taking the flow vector per pixel, a 10-bin histogram of vector orientations was computed. All the vectors in the bin with the maximum vector count were separated. The average, direction and magnitude of this vector set were taken to be the direction and magnitude of the object motion vector, $\hat{w_n}$. In other words, with respect to forward flow, $\hat{w_n}$ is the direction and amount of motion the object mask underwent to reach its new position in the following frame. Formally, if

$$\hat{w_n} = \overline{w_n(\max_w(hist(w_n))} \tag{11}$$

where $\hat{w_n}$ is the direction vector, then any n^{th} frame in the set of frames L gives an estimate of the mask position in the following frame by,

$$m^{ini}_{f_{n+1}} = m^{ini}_{f_n} + \hat{w_n} \tag{12}$$

By progressively estimating all the frames in the forward direction, m^{ini} was computed.

3.4 Object mask scale adjustment (m)

Given the object mask in the first frame m_{f_1} and the last frame m_{f_L} of a subproblem set, with m^{ini} already computed, the object mask scale adjustment objective was to determine the final m, where

$$m = (m_{f_1}, m_{f_2}, \ldots, m_{f_{L-1}}, m_{f_L}) \tag{13}$$

Here, it should be noted that $m_{f_1} = m^{ini}_{f_1}$ and $m_{f_L} = m^{ini}_{f_L}$. Hence, the task is to determine m in the remaining $L-2$ frames, from m_{f_2} till $m_{f_{L-1}}$.

A moving object, while in motion inside the FoV, might exhibit a considerable change in depth, perspective and in size. The contour encapsulating a moving object in the first frame might increase or decrease drastically in size and shape in the last frame. An important detail for object mask scale adjustment was to estimate the correspondence of each point on the object contour in the first frame with each point on the object contour in the last frame. As one-to-one correspondence was not possible, there were two options. One was to add or decrease points along the contour from the first frame until the last. This method can result in inaccuracies at each step resulting in error accumulation. Second one was to find a registration between object contours. For this purpose, the point set registration method presented in [26,27], defined by a function g, was used here. A coherent point drift (CPD) of all the points on the contour in the first frame with reference to the contour in the last frame, was estimated. A 'non-rigid' point drift estimation option was selected, as in some cases perspective changes result in self-occlusion by the object. In this case, the rigidity constraint fails to register the two contours correctly. Hence,

$$m^{CPD}_{f_L} = g(contour(m_{f_1}), contour(m_{f_L})) \tag{14}$$

As we get m_{f_1} and $m^{CPD}_{f_L}$ in the same estimated reference, the difference between the two contours was computed to estimate the *linear shape adaptation*, defined as:

$$\kappa = (m^{CPD}_{f_L} - m_{f_1})/L \tag{15}$$

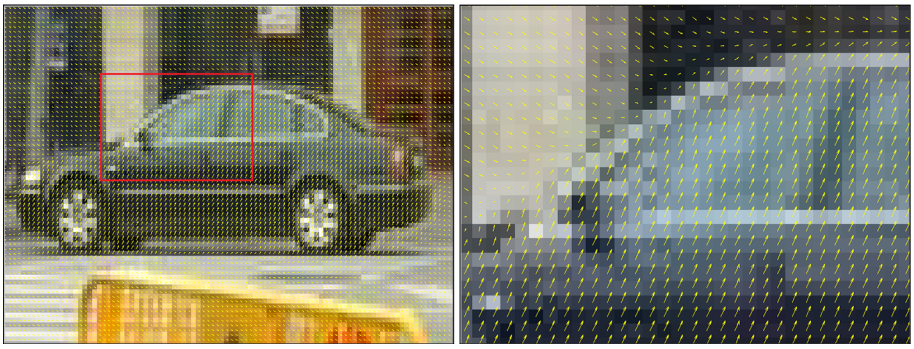

Fig. 5 Left: LDOF vectors overlay on the first frame of a moving object. The direction of flow vectors on the moving object is different from that of the background. More visible in the zoomed image on the 'Right'. Right: A zoomed image of the red bounding box from the 'Left' image. Optical flow vectors maintain consistent direction inside the car, but around the object motion boundary and on the background, the vector directions are different (color figure online)

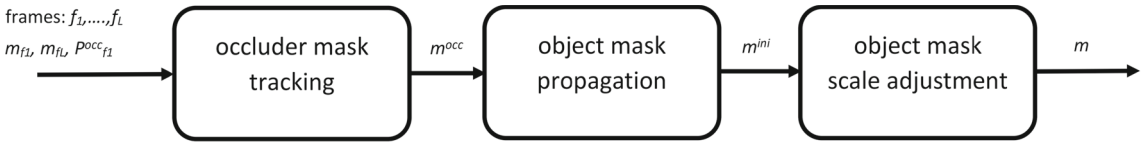

Fig. 6 Algorithmic block diagram of the functions of the motion-region annotation algorithm having one moving object pertaining to the type-2 subproblem with frame length L

Using κ, the scale of all the $L-2$ frames in the subproblem set can be adjusted, for all n by,

$$m_{f_n} = \kappa(L-n+1) * m^{ini}_{f_n} \tag{16}$$

This adjustment yields the final output m, which gives a shape estimate for the moving object, subtracting the occluder mask, on all the L frames in the subproblem set. An overall flow of the algorithm is given in Algorithm 1. The block diagram of the overall algorithm is given in Fig. 6.

This final output m is essentially the annotation mask of the object estimated for the subproblem with distortion (type 2), where [23] failed. Hence, our proposal along with the existing methodology in [23] gives forth a framework, where any object can be annotated semiautomatically with minimum user intervention. Moreover, the given proposal is able to provide an estimated ground-truth annotation in all the frames in the presence of occlusion, change in scale and perspective distortion.

All the algorithm resources including the subproblem sequences, evaluation source codes, results and the related documentation are publicly available at http://dixie.udg.edu/anntool/.

3.4.1 Complexity analysis

There are three sequential functional blocks of the algorithm. Each block contains a single for loop with the index n. The max length of the index in each case is less than total frame

Algorithm 1 Motion-Region Annotation Across Occluders

Inputs: Object: $frames \rightarrow \{f_1,\ldots,f_L\}, m_{f_1}, m_{f_L}$
Occluder: $P^{occ}_{f_1}$
Outputs: $m = \{m_{f_1}, m_{f_2},\ldots,m_{f_L}\}$

function m^{occ} = OCCLUDER MASK TRACKING($f_1,\ldots,f_L, P^{occ}_{f_1}$)
  where $m^{occ} = \{m^{occ}_{f_1}, m^{occ}_{f_2},\ldots,m^{occ}_{f_{L-1}}, m^{occ}_{f_L}\}$
  $m^{x-d}_{f_{1_{occ}}} = region(countour(\mathbf{P}_{occ_{f_1}}))$
  for $n = 1 : L-1$
    $P^{occ}_{f_n - f_{n+1}} = PointTrackerEst(\mathbf{P}^{occ}_{f_n})$
    $F_{n+1} = SURF(region(countour(\lambda \mathbf{P}^{occ}_{f_n - f_{n+1}})))$
    $T_s = SimilarityTransform(FeatureMatching(F_n, F_{n+1}))$
    $P^{occ}_{f_{n+1}} = T_s * \mathbf{P}^{occ}_{f_n}$
    $m^{occ} = region(countour(\mathbf{P}^{occ}_{f_n}))$

function m^{ini} = OBJECT MASK PROPAGATION($f_1,\ldots,f_L, m_{f_1}, m_{f_L}, m^{occ}$)
  where $m^{ini} = \{m^{ini}_{f_1}, m^{ini}_{f_1},\ldots,m^{ini}_{f_{L-1}}, m^{ini}_{f_L}\}$
  $m^{ini}_{f_1} \leftarrow m_{f_1}$; $m^{ini}_{f_L} \leftarrow m_{f_L}$;
  for $n = 2 : L-1$
    $m^{ini}_{f_n} = m^{ini}_{f_n} - (m^{ini}_{f_n} \cap m^{occ}_{f_n})$
    $w_n = \overline{LDOF(f_n, f_{n+1})}$
    $\hat{w}_n = w_n(\max_w(hist(w_n)))$
    $m^{ini}_{f_{n+1}} = m^{ini}_{f_n} + \hat{w}_n$

function m = OBJECT MASK SCALE ADJUSTMENT($f_1,\ldots,f_L, m^{ini}$)
  where $m = (m_{f_1}, m_{f_2},\ldots,m_{f_{L-1}}, m_{f_L})$
  $m_{f_1} \leftarrow m^{ini}_{f_1}$; $m_{f_L} \leftarrow m^{ini}_{f_L}$;
  $m^{CPD}_{f_L} = g(contour(m_{f_1}), contour(m_{f_L}))$
  $\kappa = (m^{CPD}_{f_L} - m_{f_1})/L$
  for $n = 2 : L-1$
    $m_{f_n} = \kappa(L-n+1) * m^{ini}_{f_n}$

length L of the subproblem. Within each *'for'* loop, the tasks are being performed in constant time. Therefore, the complexity of each functional task is $O(L)$. As these tasks are sequential, the complexity per functional block is added. Hence, the total complexity of Algorithm 1 is three times $O(L)$, which is still $O(L)$ as in terms of asymptotic complexity, $O(L)$ is sufficient to exhibit linear complexity.

4 Experimental metrics

Firstly, this section presents the evaluation methods and experimental setup used to assess the motion-region annotation result. Afterward, the performance of our proposal is exhaustively evaluated, showing both quantitative and qualitative results.

4.1 Evaluation method

The choice of evaluation criteria is such that a critical insight into the performance of the algorithm can be extracted. There are two factors at play in the motion-region annotation performance assessment: spatial and temporal. So, the goal of the criteria is to determine how accurately was the annotation propagated in terms of spatial precision as well as temporal evolution.

Spatially, the annotated region in each frame is compared with its respective ground-truth to compute the segmented region overlap performance. This, when accumulated overtime for all frames, gives an average measure of performance. This spatial performance commonly adjudged by *F-score* (F) and *Dice* (D), which are actually equivalents of each other. Sensitivity and precision are used to calculate F-score using the Hungarian method as in [48]. F(D) per frame and their average over the set of frames gives a good estimate on how well the resultant annotated region aligns with the reference. The variation in alignment over time and its reasons are, however, not addressed by these metrics. The values range from 0 to 1. With 0 being worst annotation and 1 meaning that the motion region coincides perfectly with the ground-truth.

The temporal insight on the evolution of motion-region annotation per frame is grasped profoundly by three more measures, annotated-reference region overlap ratio, occluder-object size ratio and the change in Hausdorff distance between the reference and annotated regions per frame over time.

The *annotated-reference region overlap ratio*, r_n^{a-r} is given by

$$r_n^{a-r} = \frac{m_{f_n}}{m_{GT_n}} \quad (\forall n = 1, 2, \ldots L) \tag{17}$$

where m_{f_n} and m_{GT_n} are the annotated and reference motion-regions per frame, respectively. This ratio gives an insight on how well the annotated region captures the true ground-truth in terms of its size, its evolution in time exhibits the capability of the algorithm to cope with the ground-truth even if the annotation is corrupted in the middle frames. Its value varies between 0 and 1, with 0 indicating no overlap and 1 indicating complete overlap of the two masks.

The *occluder-object size ratio*, r_n^{c-b} is given by

$$r_n^{c-b} = \frac{pixels(m_{f_n}^{occ}) \cap pixels(m_{f_n})}{pixels(m_{f_n})} \quad (\forall n = 1, 2, \ldots L) \tag{18}$$

where $m_{f_n}^{occ}$ is the occluder region in each frame. This is the ratio of the overlapped area of the occluder and annotated regions, with the total annotated motion region. This measure gives an idea on how much of the motion region is occluded by the occluder. The algorithm's performance in these regions, over time, tells us how robust the algorithm is to the size of the occluder. Here, the r_n^{c-b} value varies between 0 and 1, with 0 indicating no occlusion and 1 indicating that the object is complete occluded by the occluder. It should be noted that if r_n^{c-b} becomes 1, the algorithm will fail, as it requires some part of the moving object to be visible at all times.

The *Hausdorff distance* H^{dist} between the reference and annotated regions is given as,

$$H_n^{dist} = HausDist(m_{f_n}, m_{GT_n}) \quad (\forall n = 1, 2, \ldots L) \tag{19}$$

where $HausDist$ indicates the Hausdorff distance implementation function. Intuitively, H^{dist} finds the point p from the set m_{f_n} that is farthest from any point in m_{GT_n} and measures the distance from p to its nearest neighbor in m_{GT_n}. This measure gives an insight as to how far off the worst annotated motion-region point is with respect to the ground-truth. If evaluated over time, it gives an idea of the temporal robustness as well as the reliability of the algorithm. Here, a good annotation means that the H^{dist} value is close to zero, given in pixels. A greater H^{dist} value would indicate the magnitude of misalignment of the annotated mask with the reference.

4.2 Experimental setup

The performance of the motion-region annotation algorithm was tested on a newly formed subproblem dataset. This was done by taking a total of 25 snippets from the new motion segmentation benchmark dataset [47]. These 25 video sequences are subproblems of annotation, when the moving object underwent occlusion. Among the 25 sets of frames, 20 contain static occluders, and the remaining 5 contain moving

Fig. 7 The moving objects being annotated in the given examples are captured by a green contour around them. The static occluders are shown in blue, while the moving occluders are shown in red, bounding regions around them, respectively. Top Row: Two examples of moving objects going across single static occluder. The black car in the left image has high depth, whereas the white car in the right image has low depth, near the camera, Middle row: Two examples of moving objects going across two static occluders. The left image is high depth and the right image is medium depth, Bottom row: Two examples of moving objects going across moving occluders. The left image has very high depth, and the right image has medium depth. The moving occluder in the left image is the black moving car, which occludes our desired moving object almost completely. In the right image, the moving car is occluded by moving people (color figure online)

occluders. A few examples of this motion-region annotation dataset are shown in Fig. 7.

The 20 sequences with static occluders encompass 15 with one occluder and 5 with two occluders, as listed in Table. 2. The depth of each moving object being annotated is also indexed in three categories, low, medium and high. A moving object at low depth means that the object and occluder are near the camera, so they appear big in size and may have distinct features contained in them. A high depth means that the object size is small in the field of view. In this case, the occluder might be big or small, depending upon its own depth.

Table 2 A summary of the features of the motion-region annotation dataset

Motion-region annotation dataset features				
Datasets	Dataset features			
	Total sequences	Total frames	Avg. frames	Object depth
Static occluder (one)	15	340	22.7	Lw/Md/Hg
Static occluders (two)	5	166	33.2	Md/Hg
Moving occluders	5	177	35.4	Md/Hg

Acronyms are Avg.: Average. In object depth, Lw: Low, Md: Medium, Hg: High

In addition to the 20 sequences with static occluders, 5 more sequences were taken with moving occluder. In this case, the occluder mask is already given, as these moving occluders are motion-regions of the same sequence, which have already been annotated. The moving object depth in these sequences is also listed. All these sets of frames contain a single moving occluder.

To establish the efficacy of our work, we evaluate the performance of our algorithm in comparison with other state-of-the-art contributions. The choice of methods to utilize is limited due to a number of factors, namely availability of code, applicability on the proposed scenario (ability to propagate the label across occluders and be able to recover the shape of the moving object) and computational time. Of the listed factors, applicability of the algorithms in our scenario is a limiting factor as most algorithms fail, when motion label is propagated across an occluder. There are tracking algorithms, which are able to perform this task but they provide bounding boxes on the moving object instead of moving object boundary. Hence, we present a comparative analysis with two recent methods, a probabilistic method [16] and a learning-based method [17]. Both are moving object segmentation methods, which give the moving object motion boundary as the output. These methods do not start with known initial object boundary as in our method, so to make it fair to them, we consider their results correct on any motion they were able to correctly segment around the ground-truth. This consideration gives an advantage to the algorithms in terms of motion estimation on or around the moving object, but in effect makes them not applicable for moving object occluder sequences. Furthermore, as these methods do not estimate occluder boundary separately, hence the *occluder-object size ratio*, r_n^{c-b} is not calculated for them.

A 64-bit Intel i7 core 3.4 GHz machine with 16GB RAM was used for processing, except LDOF calculation, which was run in a similar server machine with 128GB RAM. All the scripts and results related to the experiments done are publicly available online with the motion-region dataset. Also, the scripts are designed as such to be able to incorporate any new algorithm for standardized comparison of results.

4.3 Quantitative results

The results of the motion-region annotation algorithm on the presented dataset are given in Tables 3, 4 and 5. The accumulative average F-score on static occluders as well as moving occluders reaches up to 95%.

Upon static object occlusion, a maximum F-score of 98% is achieved for *seq-03*, and the lowest is 73% for *seq-17*, where, on average, 21% motion-region area was occluded by 2 occluders, as given by the corresponding r^{c-b}. For moving occluders, a maximum F-score of 97% is achieved for *seq-24*, even in the presence of 58% occlusion. The lowest F-score in moving occluders is 94%, which is achieved even when, on average, 68% of the motion region was occluded. Observing the results in Table 4, it is observed that the overall performance of the two Probabilistic [16] and Learning-based [17] algorithms is not suitable to be used as ground-truth. In general, these algorithms do an acceptable motion segregation when the motion is small and the moving object depth in the scene is relatively small. The algorithms fail when the object is too large or too small.

The occluder-object overlap ratio, r^{c-b}, indicates the percentage amount of annotated motion-region being occluded. A higher value of this ratio signifies that the most part of the moving object is covered. It can be seen from the results that even with high r^{c-b}, the algorithm is able to propagate the label correctly in the following frames. In sequences *seq-07, seq-19, seq-22, seq-24, seq-25*, where the occlusion percentage reaches 44%, 38%, 38%, 58% and 88% respectively, the algorithm performs as high as 97% and never goes below 86%.

The annotation-reference overlap ratio r^{a-r} and Hausdorff distance H^{dist} should be understood in conjunction. r^{a-r} gives a measure of how much of the propagated annotation conforms correctly with the ground-truth, while H^{dist} measures how far the worst propagated label is from the ground-truth annotation. Here, with static occluders, it can be seen that maximum r^{a-r} of 96% is achieved in *seq-03* with H^{dist} as low as 0.03 pixels on average. The lowest overlap of 57% is experienced in *seq-17* with H^{dist} as high as 3.64 pixels on average. It is interesting to note that these results are consistent with the performance exhibited by F-score.

Table 3 A summary of the results of the label propagation algorithm on the motion-annotation dataset featuring static occluders

Results on motion-region annotation dataset having static occluders							
Seq. attributes		Spatial importance			Temporal importance		
Name	Frames	S	P	F(D)	r_n^{a-r}	r_n^{c-b}	H^{dist}
One static occluder							
seq01	24	0.94	0.96	0.95	0.90	0.01	0.29
seq02	15	0.87	0.97	0.92	0.84	0.09	0.26
seq03	29	0.98	**0.98**	**0.98**	**0.96**	0.05	**0.03**
seq04	21	**0.99**	0.95	0.97	0.94	0.06	0.12
seq05	15	**0.99**	0.97	**0.98**	0.95	0.03	0.07
seq06	20	0.97	0.95	0.96	0.92	0.04	0.10
seq07	15	0.96	0.95	0.95	0.91	**0.44**	0.14
seq08	14	0.98	0.94	0.96	0.92	0.12	0.12
seq09	15	**0.99**	0.91	0.95	0.91	0.07	0.26
seq10	73	0.93	0.93	0.93	0.87	0.07	0.14
seq11	20	0.96	0.95	0.95	0.91	0.21	0.10
seq12	19	0.93	0.96	0.95	0.90	0.17	0.15
seq13	18	0.95	0.96	0.96	0.91	0.11	0.11
seq14	21	0.92	0.88	0.90	0.82	0.26	0.52
seq15	21	0.98	0.89	0.94	0.88	0.09	0.45
Two static occluders							
seq16	25	0.93	**0.96**	0.94	0.89	0.11	0.19
seq17	20	0.72	0.73	0.73	0.57	0.21	3.64
seq18	62	**0.98**	0.91	0.94	0.89	0.03	0.40
seq19	38	0.91	0.81	0.86	0.75	**0.38**	0.48
seq20	21	0.97	0.95	**0.96**	**0.93**	0.08	**0.10**
Overall cumulative results with static occluders							
Average	25.3	0.96	0.93	0.95	0.90	0.08	0.38
Max	73	0.99	0.98	0.98	0.96	0.44	3.64
Min	14	0.72	0.73	0.73	0.57	0.01	0.03

Acronyms are Seq.: Sequences, S.: Sensitivity, P: Precision, F: F-score, D: Dice score. The best value of each metric across all sequences is bolded

One thing which cannot be appreciated through these average performance measures is the capability of the algorithm to recover, in case of failure in the intermediate frames. A temporal evaluation per frame gives a better insight on this behavior. This temporal evaluation is shown in Fig. 8, where the evolution of H^{dist} and r^{c-b} of some selected sequences per frame can be visualized.

In the figure, the temporal progress of H^{dist} and r^{c-b} in subproblem sets of frames from five video sequences are shown. It can be seen that in *seq-02* and *seq-03* as the percentage occlusion of the object, r^{c-b}, remains below 20%, then the farthest point of the annotated labeled contour from the reference label contour, H^{dist}, never increases more than 0.3 pixel. As r^{c-b} increases to almost 32% in *seq-14*, the maximum propagation error in terms of distance stays within 1.5 pixel distance. It can also be appreciated that in *seq-07* where even with a 70% peak r^{c-b}, the H^{dist} never goes beyond 0.25 pixels. This trend is also observed in the moving occluder sequence *seq-25*, where even in the presence of 88% peak r^{c-b}, the annotation error in term of H^{dist} remains within 0.5 pixels for all frames. In general, the algorithm performs well in all the sequences even in the presence of high percentage of occlusion of the moving object. Only, *seq-17* behaves differently, where in the presence of 60% occlusion, which is less than that of *seq-07* and *seq-25*, the maximum H^{dist} goes up to 11 pixels.

Another perspective of evaluation is to observe the performance of the algorithm on relatively long set of sequences. Taking one from each type, we see that in *seq-10*, *seq-18* and *seq-23* with 73, 62 and 58 frames, respectively, the algorithm had an average F-score of 95% and an average H^{dist} of 0.32 pixels. These sequences exhibit a variety of characteristics, where the moving object is, at a high depth in *seq-10*, at a medium depth in *seq-18* going across two occluders and at a medium depth in *seq-23* going across moving non-rigid occluders. The average performance shows that the algorithm

Table 4 A summary of the comparative results of metrics on the motion-annotation dataset featuring static occluders

Results on motion-region annotation dataset having static occluders									
	Probabilistic [16]			Learning [17]			Ours		
Name	F(D)	r_n^{a-r}	H^{dist}	F(D)	r_n^{a-r}	H^{dist}	F(D)	r_n^{a-r}	H^{dist}
One static occluder									
seq01	0.78	0.64	1.35	0.79	0.65	1.29	**0.94**	**0.89**	**0.19**
seq02	**0.74**	**0.59**	2.96	**0.74**	**0.59**	**2.17**	0.73	0.57	3.64
seq03	0.88	0.79	1.17	0.82	0.69	2.10	**0.95**	**0.90**	**0.29**
seq04	0.82	0.69	1.21	0.78	0.64	1.23	**0.92**	**0.84**	**0.26**
seq05	0.87	0.77	1.56	0.81	0.68	2.31	**0.98**	**0.96**	**0.03**
seq06	0.45	0.29	23.33	0.85	0.73	1.46	**0.97**	**0.94**	**0.12**
seq07	0.00	0.00	–	0.84	0.72	3.00	**0.98**	**0.95**	**0.07**
seq08	0.41	0.26	13.84	0.83	0.72	1.54	**0.96**	**0.92**	**0.10**
seq09	0.70	0.54	4.84	0.81	0.69	1.81	**0.94**	**0.89**	**0.40**
seq10	0.63	0.45	3.80	0.75	0.60	2.43	**0.95**	**0.91**	**0.14**
seq11	0.51	0.34	4.94	0.53	0.36	7.32	**0.86**	**0.75**	**0.48**
seq12	0.79	0.65	3.86	0.73	0.58	3.31	**0.96**	**0.92**	**0.12**
seq13	0.74	0.58	6.56	0.78	0.64	3.73	**0.95**	**0.91**	**0.26**
seq14	0.05	0.02	24.37	0.38	0.23	13.53	**0.93**	**0.87**	**0.14**
seq15	0.48	0.32	7.05	0.44	0.28	10.86	**0.95**	**0.91**	**0.10**
Two static occluders									
seq16	0.68	0.52	4.33	0.80	0.67	1.79	**0.95**	**0.90**	**0.15**
seq17	0.85	0.74	0.84	0.77	0.62	2.03	**0.96**	**0.91**	**0.11**
seq18	0.80	0.67	1.88	0.77	0.62	2.67	**0.96**	**0.93**	**0.10**
seq19	0.61	0.44	5.84	0.64	0.47	3.78	**0.90**	**0.82**	**0.52**
seq20	0.75	0.60	5.60	0.71	0.56	7.36	**0.94**	**0.88**	**0.45**
Overall cumulative results with static occluders									
Average	0.63	0.50	6.28	0.73	0.59	3.79	0.93	0.88	0.38
Max	0.88	0.79	24.37	0.85	0.73	13.53	0.98	0.96	3.64
Min	0.00	0.00	0.84	0.38	0.23	1.23	0.73	0.57	0.03

Acronyms are Seq.: Sequences, F: F-score, D: Dice score, r_n^{a-r}: Annotated-reference region overlap ratio and H^{dist}: Hausdorff distance. The best value for each metric against each sequence is bolded

Table 5 A summary of the results of the label propagation algorithm on the motion-annotation dataset featuring moving occluders

Results on motion-region annotation dataset having moving occluders							
Seq. attributes		Spatial importance			Temporal importance		
Name	Frames	S	P	F(D)	r_n^{a-r}	r_n^{c-b}	H^{dist}
seq21	42	**1.00**	0.92	0.96	0.92	0.12	0.14
seq22	25	0.94	0.98	0.96	0.93	0.38	**0.08**
seq23	58	0.94	0.94	0.94	0.89	0.29	0.33
seq24	36	0.95	**0.99**	**0.97**	**0.94**	0.58	**0.08**
seq25	16	0.94	0.95	0.94	0.89	**0.68**	0.14
Overall cumulative results with moving occluder							
Average	33.2	0.95	0.94	0.95	0.90	0.30	0.16
Max	58	1.00	0.99	0.97	0.94	0.68	0.33
Min	16	0.94	0.92	0.94	0.89	0.12	0.08

Acronyms are Seq.: Sequences, S.: Sensitivity, P: Precision, F: F-score, D: Dice score. The best value of each metric across all sequences is bolded

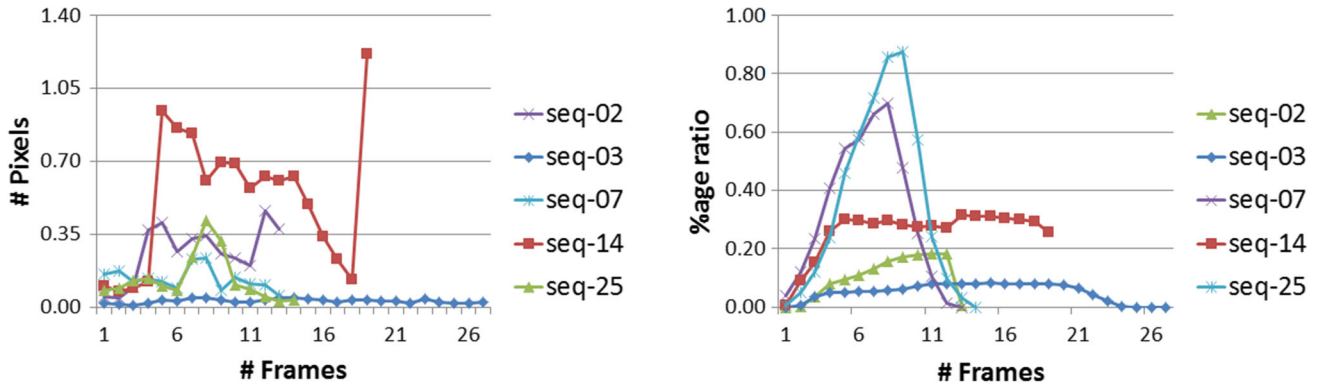

Fig. 8 The temporal evolution of performance measures of three sequences, *seq-02*, *seq-03*, *seq-07*, *seq-14* and *seq-25*. Left: A visualization of the change in H^{dist} over time in each frame (in pixels). Right: The change in occluder-object ratio r^{c-b} over time in each frame (%age)

is not affected by the length of frames as much as the type of motion in them.

4.4 Qualitative results

The qualitative results give a visual and intuitive evaluation of the algorithm. In Figs. 9 and 10, the results of motion label propagation are shown, with one occluder, two occluders and moving occluder.

In Fig. 9, three different frames, first, middle and last, of two sequences with a single static occluder are shown. In the top row from *seq-05*, a large truck is seen going across a direction post. The truck has a low depth in the field of view, meaning it is close to the camera. The average occlusion percentage is 3%, but the issue to note is that the whole body of the moving object undergoes occlusion at least once during the complete motion. The occluder mask was created with a few clicks around the direction post and it was tracked as mentioned in Sect. 3.2. It can be seen that from the start until the end, the occluder mask is robustly tracked. This robust result facilitates the shape propagation of the motion mask across all frames. As the shape and perspective change of the moving object is minimal, the results achieved are as good as 98%.

In the bottom row of the figure, three frames of the sequence *seq-08* are shown. The white car undergoes an occlusion by a tree stem. The car moves across multiple frames coming toward the camera, which changes its depth. This can be verified from the first and the last frame, as the size variation of the car is visually apparent. The thin tree stem occluder is marked in the first frame by defining a few points around it. Here, it can be seen that the area around the trunk is also marked. As the tree trunk is quite thin, the soil area around the trunk reinforces the SURF feature extraction and matching, resulting in a better tracked occluder. The linear change adaptation factor κ, as explained in Sect. 3.4, gives a good estimate of the change in depth of the car in each progressive frame. So even in the case of depth change, the achieved F-score is 96%.

In the top row of Fig. 10, three frames from *seq-19*, where the moving object is occluded by two occluders, are shown. It can be seen that the white car gets occluded by a lamp post and a thin tree trunk. Over the course of the motion, the size of the moving object changes considerably as it moves toward the camera. The occluder masks are marked in the first frame of the sequence, and it can be seen that the masks are well tracked even until the end. The object starts moving from a high depth and comes toward the camera to medium depth. With such a big change in depth, and even with 38% occlusion on average, a F-score of 86% is achieved. Here, it can be appreciated that the algorithm possesses the capability to map a small contour in the starting frames to an expanded large contour in the ending frames with consistency in shape, and vice versa.

In the middle row of Fig. 10, three frames from *seq-25*, where the moving object is occluded by a single moving occluder, are shown. An extreme case is present in this sequence, as the moving object is at a higher depth and has a small size, as compared to the moving occluder, which is at a low depth, hence quite large in size. On average the occlusion ration reaches up to 88%. Even in the presence of such occlusion, due to reliable LDOF calculation, as mentioned in Sect. 3.3, our algorithm performs well, achieving 94% F-score. Here, the moving occluder is assumed to have been previously annotated; therefore, the occluder mask marking and tracking is not performed.

In the bottom row of Fig. 10, we also show three frames from *seq-17*, where the moving object is occluded by two occluders. It can be seen that the black car goes across two lamp posts. The occluder masks are marked in the first frame and tracked until the last. In the last frame, the tracker losses the shape of a marker but it does not affect the result as there is no overlap between the wrongly tracked occluder mask and the moving object mask. Besides the occluder mask,

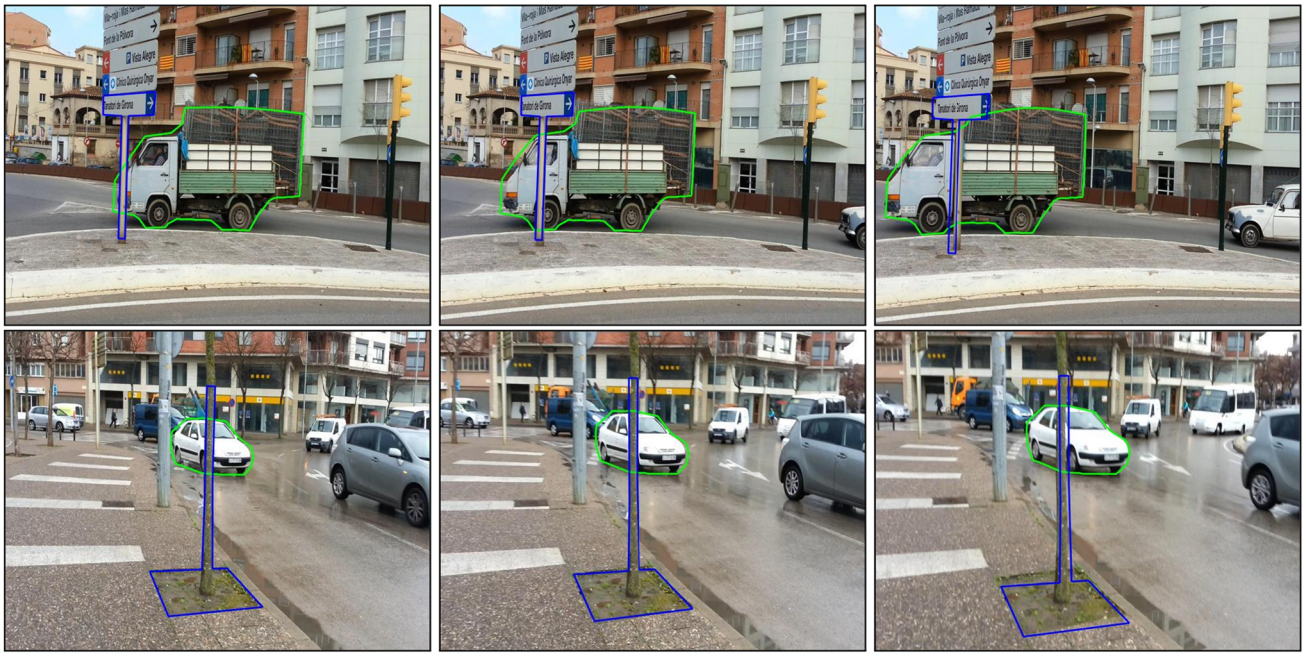

Fig. 9 Motion annotation result on three frames of two sequences containing single static occluder. The motion and the occluder masks are shown in green and blue contours, respectively. Top row: Frame 4, 9 and 14 from the sequence *seq-05*. The moving truck is occluded by the static direction post (F-score: 98%). Bottom row: Frame 2, 8 and 13 from the sequence *seq-08*. The white car is occluded by the static thin tree trunk (F-score: 96%) (color figure online)

the motion label propagation is shown by a green contour around the black car failing to propagate the label correctly. The propagated labels move ahead of the ground-truth, this means that the maximal velocity count consensus is making the mask move in the right direction but not with the correct magnitude. Upon further investigation on the obtained results, we observed that there are two competing hypothesis on the magnitude of the motion vector. Here, the wrong hypothesis edges past the correct one with a small difference. This occurs due to the background around the car because the LDOF calculated at the edges of the car gets tampered due to the color similarity between the car and the background. This limitation could be overcome by introducing a factor catering for background similarity in the maximal vector consensus.

5 Discussion

The proposed methodology is a contribution in motion annotation frameworks, where moving object labels can be propagated across occlusions. In this section, a few suggestions are proposed as future directions, which can improve the accuracy and precision even further and solve some of the limitations.

A natural progression of the framework is to develop more sophisticated methods for occluder mask shape tracking. The current method is suitable for rigid shapes with affine transformations. An occluder undergoing nonlinear change, like perspective or radial distortion, would be badly tracked by this methodology, as the overall result is sensitive to its shape tracking. A recently proposed shape tracking algorithm [49] might yield better results, as it takes a coarse to fine region-based Sobolev descent approach [49].

An enhancement in the object mask propagation approach is needed to deal with non-rigid motion masks. Currently, the motion mask is restricted to being rigid, which is good enough to cater for a lot of real motions but not all. To deal with non-rigid motion masks, the recently proposed scheme of minimal basis subspace based rigid and non-rigid segmentation approach [50] coupled with occlusion–disocclusion segregation [49] can be used in a motion model specific framework to yield acceptable results. A drawback of using image segmentation approaches for moving objects is that based on the number of frames in a video sequence the computational cost multiplies. In comparison, our approach yields quick results depending upon how fast LDOF is being calculated.

In addition to revising the object mass propagation approach for non-rigid moving objects, a nonlinear scaling adaptation factor can further improve the annotation result on a fine scale. One way of doing it would be to perform forward and backward propagation of the object mask, and then devise a cost function to penalize the non-homogeneous region overlap of the mask with the image.

Fig. 10 Motion annotation result on three frames of three sequences is shown. The motion mask is shown in green contours. The static occluder masks are shown in blue, while the moving occluder mask is shown in red, contours. Top row: Frame 2, 18 and 37 from the sequence *seq-19*. The moving white car is occluded by two static occluders, a lamp post and a tree trunk (F-score: 86%). Middle row: Frame 2, 8 and 15 from the sequence *seq-25*. The gray car is occluded by the moving black car (F-score: 94%). Bottom Row: Frame 3, 10 and 19 from the sequence *seq-17*. The black car is occluded by the two static lamp posts (F-score: 73%) (color figure online)

Assuming that the homogeneous region is part of the object and the non-homogeneous region corresponds to the background, a piece-wise fine scale adjustment of the object mask contour can be done. The objective function in such an approach would be nonlinear and computationally expensive, but the results could improve. The improvement of results using this nonlinearity, might be more apparent in sequences with a very big change in size, scale or perspective.

More recently, deep learning approaches have been a success in almost all the domains they have been applied on. On the same lines, a recent object detection and segmentation approach based on convolutional neural networks (CNNs) [51] exhibits excellent results. Our approach applied with integration of CNNs based object recognition methodologies [52,53] may yield improved results. These too would work at an exceptionally high computational cost, with a disadvantage of training and testing cycle as necessary for these approaches.

6 Conclusion

In this paper, a framework to address the problem of motion annotation in the presence of occlusion, depth change and perspective distortion has been presented. Our approach is integrated with an existing methodology [23] to formulate a framework to overcome the prevailing limitations. It was shown that with minimum manual intervention and with best utilization of the expert-time, the generation of ground-truth label for moving objects can be done even in the presence of real distortions. A three-pronged approach was taken where first the occluder mask was tracked in subsequent windows,

with SURF feature matching and similarity transformation. Then, the object mask propagation was done by computing maximal consensus motion vectors from the state-of-the-art LDOF estimation. And finally, the scale adjustment of the propagated object mask was performed by first to last frame point-set registration couple with linear adaptation factor κ. For evaluation, we also presented a motion annotation dataset with 25 sequences, containing single and multiple static and moving occluders. We presented a detailed quantitative and qualitative analysis of the methodology to show that it can be reliably used for label propagation in sequences with occlusion and other real noises, reaching an average F-Score as high as 95%. In addition, we performed a comparative analysis of our proposal with two state-of-the-art methodologies. We have also shared the source codes, results and the related documentation publicly for the community to use it and to perform further improvements in this methodology.

Acknowledgements This work is supported by the project NICOLE (Ref TIN2014-55710-R) and also by MPC UdG 2016/022 Grant. Muhammad Habib Mahmood is supported by an FI Grant. Yago Diez is supported by the IMPACT Tough Robotics Challenge Project of Japan Science and Technology Agency. Prof. X. Lladó is supported by the ICREA Academia program.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

References

1. Song, D., Kim, C., Park, S.-K.: A multi-temporal framework for high-level activity analysis: violent event detection in visual surveillance. Inf. Sci. **447**, 83–103 (2018)
2. Huerta, I., Pedersoli, M., Gonzàlez, J., Sanfeliu, A.: Combining where and what in change detection for unsupervised foreground learning in surveillance. Pattern Recogn. **48**(3), 709–719 (2015)
3. Kushwaha, A.K.S., Srivastava, R.: A framework of moving object segmentation in maritime surveillance inside a dynamic background. J. Comput. Sci. 35–54 (2015)
4. Ali, M.N., Abdullah-Al-Wadud, M., Lee, S.-L.: Multiple object tracking with partial occlusion handling using salient feature points. Inf. Sci. **278**, 448–465 (2014)
5. Wei, L., Wang, X., Yin, J., Wu, A.: Self-regularized fixed-rank representation for subspace segmentation. Inf. Sci. **412**, 194–209 (2017)
6. Kc, A.K., Jacques, L., De Vleeschouwer, C.: Discriminative and efficient label propagation on complementary graphs for multi-object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **39**(1), 61–74 (2017)
7. Chen, B.-J., Medioni, G.: Exploring local context for multi-target tracking in wide area aerial surveillance. In: IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 787–796 (2017)
8. Rubino, C., Crocco, M., Murino, V., Del Bue, A.: Semantic multi-body motion segmentation. In: IEEE Winter Conference on Applications of Computer Vision, pp. 1145–1152 (2015)
9. Liu, W., Lau, R.W., Manocha, D.: Robust individual and holistic features for crowd scene classification. Pattern Recogn. **58**, 110–120 (2016)
10. Li, Y., Wang, X., Liu, W., Feng, B.: Deep attention network for joint hand gesture localization and recognition using static RGB-D images. Inf. Sci. **441**, 66–78 (2018)
11. Wu, D., Pigou, L., Kindermans, P., Le, N., Shao, L., Dambre, J., Odobez, J.: Deep dynamic neural networks for multimodal gesture segmentation and recognition. IEEE Trans. Pattern Anal. Mach. Intell. **38**(8), 1583–1597 (2016)
12. Mademlis, I., Tefas, A., Pitas, I.: A salient dictionary learning framework for activity video summarization via key-frame extraction. Inf. Sci. **432**, 319–331 (2018)
13. Voigtlaender, P., Leibe, B.: Online adaptation of convolutional neural networks for video object segmentation, arXiv preprint arXiv:1706.09364
14. Pont-Tuset, J., Caelles, S., Perazzi, F., Montes, A., Maninis, K.-K., Chen, Y., Van Gool, L.: The 2018 Davis challenge on video object segmentation, arXiv preprint arXiv:1803.00557
15. Caelles, S., Maninis, K.-K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation, arXiv:1611.05198
16. Bideau, P., Learned-Miller, E.: It's moving! A probabilistic model for causal motion segmentation in moving camera videos. In: European Conference on Computer Vision, pp. 433–449 (2016)
17. Weinzaepfel, P., Revaud, J., Harchaoui, Z., Schmid, C.: Learning to detect motion boundaries. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2578–2586 (2015)
18. Shen, C., Chen, Y., Guan, X.: Performance evaluation of implicit smartphones authentication via sensor-behavior analysis. Inf. Sci. **430**, 538–553 (2018)
19. Yi, S., Li, H., Wang, X.: Understanding pedestrian behaviors from stationary crowd groups. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3488–3496 (2015)
20. Yang, D., Guo, J., Wang, Z.-J., Wang, Y., Zhang, J., Hu, L., Yin, J., Cao, J.: Fastpm: an approach to pattern matching via distributed stream processing. Inf. Sci. **453**, 263–280 (2018)
21. Liu, L., Wang, S., Su, G., Hu, B., Peng, Y., Xiong, Q., Wen, J.: A framework of mining semantic-based probabilistic event relations for complex activity recognition. Inf. Sci. **418**, 13–33 (2017)
22. Zhang, Y., Lu, H., Zhang, L., Ruan, X.: Combining motion and appearance cues for anomaly detection. Pattern Recogn. **51**, 443–452 (2016)
23. Liu, C., Freeman, W.T., Adelson, E.H., Weiss, Y.: Human-assisted motion annotation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
24. Brox, T., Malik, J.: Large displacement optical flow: descriptor matching in variational motion estimation. IEEE Trans. Pattern Anal. Mach. Intell. **33**(3), 500–513 (2011)
25. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Comput. Vis. Image Underst. **110**(3), 346–359 (2008)
26. Myronenko, A., Song, X.: Point set registration: coherent point drift. IEEE Trans. Pattern Anal. Mach. Intell. **32**(12), 2262–2275 (2010)

27. Jian, B., Vemuri, B.C.: Robust point set registration using gaussian mixture models. IEEE Trans. Pattern Anal. Mach. Intell. **33**(8), 1633–1645 (2011)
28. Doermann, D., Mihalcik, D.: Viper: tools and techniques for video performance evaluation applied to scene and document images. In: Symposium on Document Image Understanding Technology, p. 339 (2001)
29. Kavasidis, I., Palazzo, S. Di Salvo, R., Giordano, D., Spampinato, C.: A semi-automatic tool for detection and tracking ground truth generation in videos. In: International Workshop on Visual Interfaces for Ground Truth Collection in Computer Vision Applications, p. 6 (2012)
30. Kavasidis, I., Palazzo, S., Di Salvo, R., Giordano, D., Spampinato, C.: An innovative web-based collaborative platform for video annotation. Multimed. Tools Appl. **70**(1), 413–432 (2014)
31. Yuen, J., Russell, B., Liu, C., Torralba, A.: Labelme video: building a video database with human annotations. In: IEEE International Conference on Computer Vision, pp. 1451–1458 (2009)
32. Bianco, S., Ciocca, G., Napoletano, P., Schettini, R.: An interactive tool for manual, semi-automatic and automatic video annotation. Comput. Vis. Image Underst. **131**, 88–99 (2015)
33. Mahmood, M.H., Salvi, J., Lladó, X.: Semi-automatic tool for motion annotation on complex video sequences. Electron. Lett. **52**(8), 602–604 (2016)
34. Badrinarayanan, V., Galasso, F., Cipolla, R.: Label propagation in video sequences. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3265–3272 (2010)
35. Lin, G., Liao, K., Sun, B., Chen, Y., Zhao, F.: Dynamic graph fusion label propagation for semi-supervised multi-modality classification. Pattern Recogn. **68**, 14–23 (2017)
36. Vondrick, C., Patterson, D., Ramanan, D.: Efficiently scaling up crowdsourced video annotation. Int. J. Comput. Vis. **101**(1), 184–204 (2013)
37. Spiro, I., Taylor, G., Williams, G., Bregler, C.: Hands by hand: crowd-sourced motion tracking for gesture annotation. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 17–24 (2010)
38. Budvytis, I., Badrinarayanan, V., Cipolla, R.: Label propagation in complex video sequences using semi-supervised learning. Br. Mach. Vis. Conf. **2257**, 2258–2259 (2010)
39. Chen, A., Corso, J.: Propagating multi-class pixel labels throughout video frames. In: Western New York Image Processing Workshop, pp. 14–17 (2010)
40. Vijayanarasimhan, S., Grauman, K.: Active frame selection for label propagation in videos. In: European Conference on Computer Vision, pp. 496–509 (2012)
41. Karasev, V., Ravichandran, A., Soatto, S.: Active frame, location, and detector selection for automated and manual video annotation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2123–2130 (2014)
42. Raheb, E., Katerina, Buccoli, M., Zanoni, M., Katifori, A., Kasomoulis, A., Sarti, A., Ioannidis, Y.: Towards a general framework for the annotation of dance motion sequences. Multimed. Tools Appl. 1–33 (2022)
43. Gao, L., Song, J., Nie, F., Yan, Y., Sebe, N., Tao Shen, H.: Optimal graph learning with partial tags and multiple features for image and video annotation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4371–4379 (2015)
44. Liu, Y., Yuan, Z., Chen, B., Xue, J., Zheng, N.: Illumination robust color naming via label propagation. In: IEEE International Conference on Computer Vision, pp. 621–629 (2015)
45. In Kim, K., Tompkin, J., Pfister, H., Theobalt, C.: Context-guided diffusion for label propagation on graphs. In: IEEE International Conference on Computer Vision, pp. 2776–2784 (2015)
46. Mahmood, M.H., Zappella, L., Díez, Y., Salvi, J., Lladó, X.: A new trajectory based motion segmentation benchmark dataset (UdG-MS15). In: Iberian Conference on Pattern Recognition and Image Analysis, pp. 463–470 (2015)
47. Mahmood, M.H., Díez, Y., Salvi, J., Lladó, X.: A collection of challenging motion segmentation benchmark datasets. Pattern Recogn. **61**, 1–14 (2017)
48. Ochs, P., Malik, J., Brox, T.: Segmentation of moving objects by long term video analysis. IEEE Trans. Pattern Anal. Mach. Intell. **36**(6), 1187–1200 (2014)
49. Yang, Y., Sundaramoorthi, G.: Shape tracking with occlusions via coarse-to-fine region-based Sobolev descent. IEEE Trans. Pattern Anal. Mach. Intell. **37**(5), 1053–1066 (2015)
50. Lee, C.M., Cheong, L.F.: Minimal basis subspace representation: a unified framework for rigid and non-rigid motion segmentation. Int. J. Comput. Vis. 1–25 (2016)
51. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Region-based convolutional networks for accurate object detection and segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **38**(1), 142–158 (2016)
52. Wang, Z., Deng, Z., Wang, S.: Sam: a rethinking of prominent convolutional neural network architectures for visual object recognition. In: IEEE International Joint Conference on Neural Networks, pp. 1008–1014 (2016)
53. Alexandre, L.A.: 3d object recognition using convolutional neural networks with transfer learning between input channels. Intell. Auton. Syst. **13**, 889–898 (2016)

Muhammad Habib Mahmood received his Bachelors of mechatronics Engineering (2003) from National University of Sciences and Technology, Pakistan. He graduated from the Erasmus Mundus Masters in Vision and Robotics (Vibot) program in 2010 spending time in Heriot-Watt University (UK), University of Girona (Spain) and University of Burgundy (France). He received his PhD in Technology from University of Girona in 2018 working on computer vision and machine learning. He has since been a part of the Electrical and Computer Engineering department of Air University, Islamabad, Pakistan, in the capacity of Assistant Professor. He is also the Program Head of the Computer Engineering program. He takes keen interest in video analytics, motion analysis, generative models, attention networks and data analytics, among other areas. He is affiliated with STech.ai, an AI solution provider as an Ai team lead since 2019. He has authored more than 20 articles in peer-reviewed journals and conferences.

Yago Diéz received his B.S. degree in Mathematics (2002) from Barcelona Tech and a Ph.D. in Software Development (2008) in a joint degree from the University of Girona and Barcelona Tech. From 2008 until 2015, he was a Post-doctoral Researcher at Girona University (Spain) in the Computer Vision and Robotics Department (VICOROB). From 2015 to 2017, he worked as an Assistant Professor at GSIS Tokuyama Lab at Tohoku University (Sendai, Japan). Since 2017, he is an Associate Professor at the faculty of Science in Yamagata University (Yamagata Japan). His main current research interests are deep learning applications of natural image processing, computer vision algorithms for UAV-acquired image processing and medical image registration (with special focus on multiple sclerosis imaging, ABUS and breast MRI). Past research interests include pattern recognition, computational geometry and 3D coarse matching of point clouds. He has published 27 JCR journals and more than 50 articles in conferences with peer review.

Arnau Oliver graduated in Physics at Universitat Autònoma de Barcelona and as Technical Engineer in Computer Systems at Universitat de Girona, where he also received his Ph.D. degree in Information Technology. He is currently working as Associate Professor at the Computer Architecture and Technology Dept of Universitat de Girona, where he is also member of the Computer Vision and Robotics Research Institute. He has authored more than 200 papers in journals and proceedings of international conferences and supervised 10 PhD theses. His research interests include medical image computing and analysis, especially focused on the quantitative analysis of diseased brains and the development of automatic tools for early cancer detection.

Joaquim Salvi graduated in computer science from the Technical University of Catalonia, in 1993, received the D.E.A. (M.Sc.) degree in computer science and the Ph.D. degree in industrial engineering from the University of Girona, in 1996 and 1998, respectively. He was a Visiting Professor with the Ocean Systems Lab, Heriot-Watt University, UK. He is currently a Full Professor with a Chair of Computer Vision, Computer Architecture and Technology Department and at the Computer Vision and Robotics Group, University of Girona. He is also the Rector of the University of Girona. He is involved in some governmental projects and technology transfer contracts to industry. He has authored two books and 185 scientific articles, with an h-index of 33 and more than 7000 cites. His current interests include computer vision and image processing applied to medical imaging, machine vision and robotics. He is a Charter Member of the spinoff companies AQSense and BonesNotes. He received the Best Thesis Award in engineering for his Ph.D.

Xavier Lladó Prof. Xavier Lladó received the B.S. degree in Computer Science (1999) and the PhD in Computer Engineering (2004) from the University of Girona. From 2004 to 2006, he was working as a Postdoctoral Research Assistant in the Department of Computer Science at Queen Mary, University of London. In 2006, he moved to a Lecturer position in the Department of Computer Architecture and Technology of the University of Girona, where he is currently a Full Professor. He is also a member of the Computer Vision and Robotics (VICOROB) Research Institute, where he is leading the Advanced Image Analysis group. He is a senior member of the IEEE and an ICREA Academia member. He has published more than 280 papers in journals and conferences and supervised 13 PhD theses (2 of them receiving the best PhD award from the University of Girona). He is also cofounder and CSO of the spin-off Tensormedical of the University of Girona and Hospital Vall d'Hebron de Barcelona created in 2020.