International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

Seeing the Objects Behind the Dots: Recognition in Videos from a Moving Camera

Björn Ommer · Theodor Mader · Joachim M. Buhmann

Received: date / Accepted: date

Abstract Category-level object recognition, segmentation, and tracking in videos becomes highly challenging when applied to sequences from a hand-held camera that features extensive motion and zooming. An additional challenge is then to develop a fully automatic video analysis system that works without manual initialization of a tracker or other human intervention, both during training and during recognition, despite background clutter and other distracting objects. Moreover, our working hypothesis states that category-level recognition is possible based only on an erratic, flickering pattern of interest point locations without extracting additional features. Compositions of these points are then tracked individually by estimating a parametric motion model. Groups of compositions segment a video frame into the various objects that are present and into background clutter. Objects can then be recognized and tracked based on the motion of their compositions and on the shape they form. Finally, the combination of this flow-based representation with an appearance-based one is investigated. Besides evaluating the approach on a challenging video categorization database with significant camera motion and clutter, we also demonstrate that it generalizes to action recognition in a natural way.

This work was supported in part by the Swiss national science foundation under contract no. 200021-107636.

Björn Ommer
Department of EECS
University of California, Berkeley
E-mail: ommer@eecs.berkeley.edu

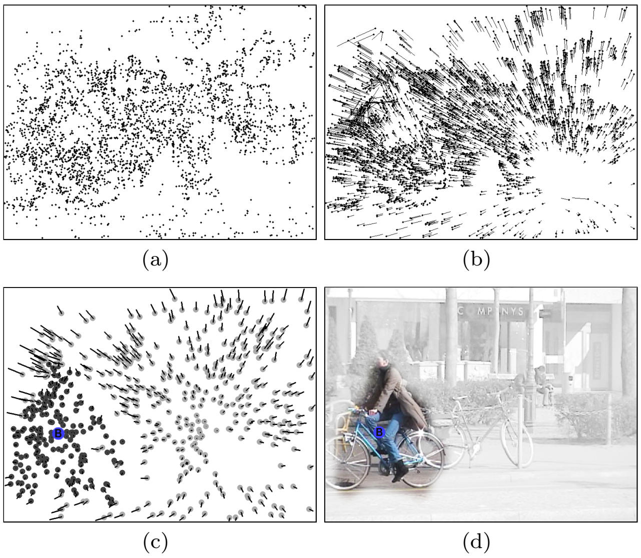
Theodor Mader · Joachim M. Buhmann
Department of Computer Science
ETH Zurich
E-mail: jbuhmann@inf.ethz.ch

Keywords object recognition · segmentation · tracking · video analysis · compositionality · visual learning

1 Introduction

Object recognition in images and videos poses a long standing key challenge for computer vision and the complexity of this problem heavily depends on the constraints and restrictions that can be imposed on the data. Significant progress has been made in scenarios of limited complexity (e.g. exemplar detection [23], fully supervised learning [10], videos that allow for background subtraction [40] as in the training of [36], etc.). However, the much more general and less constraint setting of category-level object recognition in videos from a hand-held camera (featuring motion and zoom) without heavy supervision during training still poses a highly ambitious computer vision task and the required algorithms are situated at the forefront of modern vision research. Although recent research has pushed this frontier considerably (cf. the large body of work on still image categorization), the problem in its general form still remains one of the great challenges of machine vision. When multiple complex subtasks have to be jointly solved—as in automatic video analysis—the untertaking becomes even more ambitious. The vision system described in this contribution addresses this challenging recognition problem and it also allows us to investigate if object recognition in videos is feasible based only on an erratic pattern of interest point locations without having to extract complex features (see Figure 1 for an illustration).

In order to build a complete system for video analysis, several other tasks have to be dealt with besides



(a) (b) (c) (d)

Fig. 1 Only based on interest points (a), which jitter between frames, and on optical flow (b), objects are recognized (c) despite zooming and camera panning. Therefore, compositions of points are established, tracked over time, and segmented (c), (d).

category-level recognition—most prominently segmentation and tracking. These subtasks are all directly coupled with each other: Segmentation yields object shape and is thereby directly linked to shape-based recognition, while tracking of local object regions provides the basis for segmentation. Apart from tackling these individual subproblems, our paper also significantly contributes to the systems design aspect—showing how all of these subtasks can be combined in a computer vision system so that they mutually benefit from another.

In detail, the following problems are investigated:
Category-level recognition: The basic setting is that of category-level object recognition [11] for multiple categories as opposed to single-class approaches such as the pedestrian detectors [36, 43, 8] or exemplar detection [23, 39, 44]. Categorization aims at finding all the diverse instances of a category, whereas exemplar recognition detects different views of the same object instance. Due to the large intra-class variations, categorization is generally considered to be a harder task than exemplar detection since a single object model has to comprise very diverse instances.
Reducing supervision during learning: We study if it is possible to learn the underlying object models without requiring manual annotation of object structure or labeling individual objects in a video sequence. In the field of categorization (e.g. [11]), this is typically referred to as unsupervied learning of object models. In the machine learning community this setting is commonly called weakly supervised learning since a global category label is given for the training images while detailed annotations are missing. Object categorization differs insofar as the degree of supervision typically refers to the user information provided for learning the structure of objects *within each category* (e.g. no segmentation, bounding boxes, accurate hand segmentation, or even precise labelling of object parts). Our learning algorithm requires only the category label of the most prominent object of a cluttered video sequence during training but it does not need hand segmentations or other localization information. So the structure (visual representation such as shape, appearance, etc.) of objects in each category is learned without supervision information. Also we are not pursuing a pure query-by-example approach like [39] where regions are identified that are similar to a user selected one. Therefore, our approach has to automatically discover what the relevant object information in the training samples is and separate it from clutter. Segmentation is, consequently, tightly coupled with recognition so that our approach differs from a popular trend in the field of categorization—namely models based on rigid, regular-grid-like templates with bag-of-features descriptors in each cell, e.g. [19]. Such models depend on manual supervision with bounding box segmentations during training.
Segmentation of videos from a moving camera: To learn accurate object models, many state-of-the-art recognition systems require accurate pixel-level segmentations, e.g. [21, 36]. Therefore, the setting is typically limited to static cameras or homogeneously textured backgrounds. In such restricted scenarios, background subtraction [40] suffices to separate objects from clutter. The approach to action recognition [2] avoids the segmentation problem by assuming that accurate object silhouettes are available (e.g. by background subtraction). In our much more general setting of a moving (e.g. panning or shaking) and zooming camera a difficult object segmentation problem has to be solved where objects are segregated from each other and from the background and we explicitly couple recognition with segmentation. This coupling with recognition advances our method beyond classical segmentation techniques such as layered motion approaches [45, 15]. In contrast to these methods we are computing the segmentation based on the sparse set of compositions that are used to recognize objects, rather than on the basis of a dense flow field. For the purpose of exclusive video segmentation without recognition, decompositions of segments into subregions have been studied in [18]. In our approach, the spatial arrangement of compositions, which arises from segmentation and tracking, provides the basis for shape-based recognition. Thereby, segmentation builds on the representation used for recognition

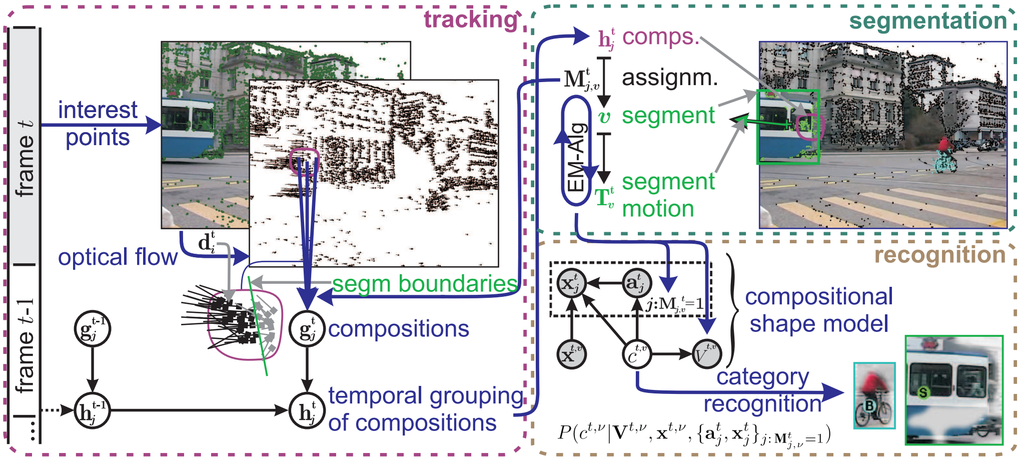


Fig. 2 Outline of the processing pipeline. Tracking groups interest points spatially and over time to form compositions and feeds these into segmentation. Segmentation feeds segments back into tracking (line 8 of Alg. 2) to prune compositions at segment boundaries. Moreover, segmentation provides object shape $\mathbf{V}^{t,\nu}$ and segment assignments of compositions for recognition.

and vice versa. Other recognition approaches with moving camera [20] use a lot of restricting side information about the scene (e.g. assuming that objects are only appearing in certain image regions) and about the relative motion of the camera with respect to the scene (based on additional sensors) that is not available in the general setting. [33] presents an interesting approach that is specifically designed for segmentation of aerial videos where frame stabilization using ground-plane homographies is possible since the background is typically very distant. Moreover, they have also experimented with generalized PCA [41,42] but obtained disappointing results which they attribute to the lack of spatial constraints in the approach by Vidal et al. Another weakness of generalized PCA is that the required number of sample points grows exponentially with the number of subspaces. This problem is dealt with in [46] by phrasing segmentation as a linear manifold finding problem which has a linear complexity. Finally, a related, but independently developed approach based on structure from motion point clouds is discussed in [4]. This approach is however restricted to the case where changes between frames are solely the result of camera motion. In contrast to this, our system operates in the general case where objects and camera can move arbitrarily. Obviously, this generalization is crucial for action recognition.

Tracking without manual interaction: Along with the video segmentation problem comes that of tracking objects from one frame into the next so that consecutive segmentations can be brought into correspondence. Powerful tracking algorithms have been proposed that track a user specified region from one frame into the next, see for instance [6,1]. Manual initialization considerably simplifies the underlying correspondence problem. Our goal is, however, to build a video analysis system that works without any manual interaction so that hand initializations of a tracker are not an option. Since our approach is not provided with an initialization on which object to track and no manual object segmentation is given, we cannot track the whole object directly but only various local regions of the scene. The task of the algorithm is then to automatically group regions that belong to the same object and to establish correspondences between segmentations of consecutive frames.

Object models and shape representation: As a result we obtain an object representation which is a compositional model in the spirit of [17,32]: Objects correspond to image segments and these consist of a variable number of compositions of simple, atomic parts. Compositions describe characteristic object regions whereas the atomic parts are local image descriptors (such as localized feature histograms [30]). However, these approaches are specifically designed for the analysis of still images and do not deal with tracking or segmentation. Moreover, [17] even excludes the question of model learning.

We then ask the following question: Is it possible to construct robust object representations only on the basis of the flow at interest points and the shape they describe? Such a model has the computational advantage that no additional appearance features have to be extracted. Moreover, it provides a natural way of dealing with object classes that are inhomogeneous in their visual appearance (e.g. pedestrians with different clothing). It turns out that we can indeed recognize objects based only on a flickering point pattern that is learned without supervision from cluttered training images. Therefore, recognition is tightly coupled with segmentation so that the shape of points in an object segment becomes discriminative. The key to solving the segmentation problem based on ambiguous local flow measurements is to restrict the space of potential transformations between frames. Finally, we also combine this flow-based shape model with the appearance-based compositional object representation of [32] to investigate the gain of two complementary representations (shape/appearance). In reference to our results presented in Section 4, we can summarize that the flow-based shape model shows already state-of-the-art performance for object recognition. Combining this representation with appearance information yields an additional performance gain. All in all we can conclude that our flow-based approach leads to robust object representations that render recognition feasible. In contrast to flow-based shape which is of course superior to a flow-based motion representation for the task of object recognition, action recognition obviously requires a representation based on motion. In Section 4.3, we comment on these findings in detail.

Contrary to [31] where we have applied histogram clustering for segmentation, this contribution uses a parametric model for segments to cope with complex flow fields (e.g. zooming). Moreover, the present method explicitly represents object shape. Lastly, the system performs in near real-time on full PAL resolution videos. We have evaluated it on a video categorization database and in addition we show that it is also suited for action recognition tasks that have been addressed by more restricted approaches in the literature.

Outline of the Processing Pipeline

Let us now summarize the processing pipeline (illustrated in Figure 2) before taking a detailed look at the individual steps in later sections. A novel video is analyzed frame-by-frame, while the underlying model establishes correspondences between consecutive frames. Processing starts by computing optical flow at those image locations where flow information can be reliably estimated. Since these interest points vary from frame to frame they cannot be tracked through a whole image sequence. We therefore group these points spatially to establish local ensembles of interest points. These compositions behave robustly with respect to individual miscalculated optical flow vectors or disappearing interest points and they can, for that reason, be reliably *tracked* throughout a video sequence. The goal is then to group compositions that belong to the same object and separate them from those compositions of other objects or background clutter. This *segmentation* problem is solved using an expectation-maximization approach and we choose a parametric representation for object segments that resides in the optical flow space, i.e. this is a flow-based segmentation procedure. After tracking and segmentation the third task that has to be addressed is *object recognition*. We therefore study different object representations—foremost those that are only based on optical flow and global object shape, but we also investigate their combination with models of local object appearance.

In the training phase, tracking and segmentation proceed as described above. The generated object representations are collected over all frames and are fed as training samples into a probabilistic classifier which learns the object model.

2 Region Tracking and Object Segmentation

2.1 Tracking Object Regions

In a first step optical flow information has to be computed in a video frame. We use the method of Shi and Tomasi [37] to find interest points (IPs) at which flow can be estimated reliably. Optical flow is then computed by tracking the interest points from the preceding frame into the next using the Lucas-Kanade tracking algorithm [24]. Let $\mathbf{d}_i^t \in \mathbb{R}^2$ denote the optical flow estimated at interest point i in frame t, i.e. the displacement vector.

Compositions as Spatial Groupings of Parts: In the initial frame of a video ($t = 0$), a random subset of all detected interest points is selected (the cardinality of this set is one-tenth of the number of interest points). Each of these points is grouped with the interest points in its local neighborhood (radius of $w = 30$ pixel chosen using cross-validation) yielding ensembles of interest points, the compositions $\mathbf{g}_j^t \in \mathbb{R}^2$. Let $\Gamma^t(j)$ denote the set of interest points in the local neighborhood of j-th composition $\mathbf{g}_j^t$,

$$\Gamma^t(j) = \{i : \text{IP } i \text{ in neighborhood of } j\text{-th comp.}\}\,. \quad (1)$$

A composition represents all its constituent interest points $i \in \Gamma^t(j)$ by calculating the mean of their flow vectors

$$\mathbf{g}_j^t := \frac{1}{|\Gamma^t(j)|} \sum_{i \in \Gamma^t(j)} \mathbf{d}_i^t\,. \quad (2)$$

We have also tested the median without observing a significant performance gain.

Tracking Compositions: The goal is then to let compositions move together with the object that they cover so that each composition can measure how its underlying object region behaves over time (i.e. how

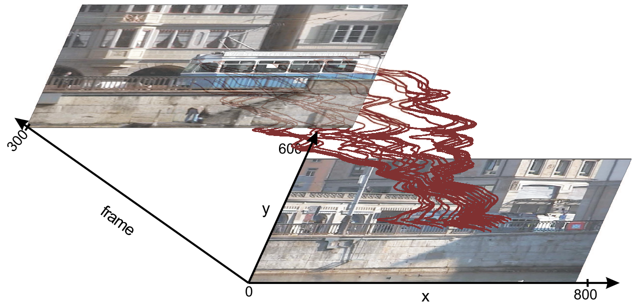


Fig. 3 Tracking compositions over 300 frames despite camera shake and panning. Only the trajectories of compositions that are assigned to the object segment are plotted.

it moves over several frames). Thus, compositions can combine information over multiple frames, which is not possible with the grid of static boxes as proposed in [26]. Given the position $\mathbf{x}_j^t \in \mathbb{R}^2$ of the j-th composition in frame t and the average optical flow of this image region $\mathbf{g}_j^t$, its predicted position in the next frame is

$$\mathbf{x}_j^{t+1} := \mathbf{x}_j^t + \frac{1}{|\Gamma^t(j)|} \sum_{i \in \Gamma^t(j)} \mathbf{d}_i^t \,. \qquad (3)$$

The initial location $\mathbf{x}_j^0$ is set to the center of all interest points that are assigned to a composition. The composition is then updated in frame $t+1$ by assigning all interest points in its local neighborhood to this composition. Consequently, the prediction (3) and the assignment step (1) are alternated once in each new frame (see Alg. 2 for details). Figure 3 shows the trajectories of compositions that are tracked despite camera shake and panning.

Temporal Grouping of Compositions: Whereas the previously presented grouping runs in the spatial domain (using proximity) the following presents a grouping of compositions over time. Forming ensembles of compositions over consecutive frames increases the robustness of the representation since measurement errors in individual frames have less influence. The temporal grouping of the j-th composition over successive frames yields temporal compositions

$$\mathbf{h}_j^t = \eta \mathbf{g}_j^t + (1-\eta)\mathbf{h}_j^{t-1} \,. \qquad (4)$$

For simplicity the weight is chosen to be $\eta = 1/2$ and $\mathbf{h}_j^1 = \mathbf{g}_j^1$. Consequently, the influence of older compositions is decaying exponentially over time.

2.2 Joint Tracking and Segmentation of Objects based on Floating Image Regions

Motivation: Typically, the object tracking problem is substantially simplified by manually initializing the tracker with an object region that is to be tracked. A common state-of-the-art approach along that line is to treat wide-baseline feature matching as a classification-based tracking problem [1,22,13], where an object model is trained on the region of interest against background clutter. The classifier then predicts the location of the object in new videos frames. Both off-line [22] and online [13] learning techniques have been applied. Moreover, distinctive features (such as SIFT [23] or the learned features of [13]) are used that simplify the matching procedure by providing distinctive matches even over a wide baseline. When highly distinctive features are lacking, object tracking is aggravated. For that reason, other purely flow based approaches [3] depend on a scenario where the background can be subtracted (e.g. static camera).

Approach: Since we have no information on where the objects are in a frame, we can only track local object regions (the tracking of compositions $\mathbf{h}_j^t$ presented in Section 2.1) as opposed to complete objects. The problem is then to assemble the various object regions into the different objects and into background before we can, finally, track complete objects. This is basically a segmentation problem where compositions have to be assigned to segments and, simultaneously, the segments have to be computed. At the same time, each segment has to be brought into correspondence with its counterpart in the previous frame, since object segments in consecutive frames are representing the same objects. Such mutually dependent problems are commonly solved by adopting an *expectation-maximization approach* [28].

Let there be $K-1$ objects plus background clutter in the scene. In [31] we have investigated how the number of objects can be automatically estimated using a stability analysis. Moreover, the model complexity can change within a video sequence. When K changes, the system switches between segmentations of different complexity (in our experiments K has varied in the range of 2 to 5). Therefore, the following discussion excludes the aspect of automatic estimation of model complexity by assuming that the correct K is provided as input. That way we hope to avoid distracting from our main theme of combined tracking, segmentation, and recognition.

Then the task of segmentation is to assign each composition $\mathbf{h}_j^t$ to a single segment $\nu \in \{1, \dots, K\}$, i.e. we have to compute the assignment matrix

$$\mathbf{M}_{j,\nu}^t = \mathbf{1}\{j\text{-th comp assigned to segm } \nu\} \in \{0,1\} \,. \qquad (5)$$

Here $\mathbf{1}\{\cdot\}$ denotes the characteristic function. Based on all assigned compositions, segments are computed. Since the samples that are being clustered are flow vectors, the segment prototypes will be transformation matrices that represent all the flow vectors in a segment. Because optical flow provides much more ambiguous correspondences between frames than highly distinctive features such as SIFT, we have to restrict the space of admissible correspondences to disambiguate the estimation problem. A reasonable assumption is that *similarity transformations* suffice to model the deformation of an object between two consecutive frames. Segments ν are then represented by a similarity transformation matrix $\mathbf{T}_\nu^t$ in homogeneous coordinates (see [14]), which is defined by the parameters α (rotation), s (scale), δ_x, and δ_y (translation),

$$\mathbf{T}_\nu^t = \begin{pmatrix} s\cos\alpha & -s\sin\alpha & \delta_x \\ s\sin\alpha & s\cos\alpha & \delta_y \\ 0 & 0 & 1 \end{pmatrix} . \tag{6}$$

This transformation matrix yields an approximation $\mathbf{T}_\nu^t\,(\mathbf{x}_j^t, 1)^\top - (\mathbf{x}_j^t, 1)^\top$ to the flow vectors $\mathbf{h}_j^t$ of compositions in segment ν:

$$\mathbf{T}_\nu^t \begin{pmatrix} \mathbf{x}_j^t \\ 1 \end{pmatrix} - \begin{pmatrix} \mathbf{x}_j^t \\ 1 \end{pmatrix} \approx \begin{pmatrix} \mathbf{h}_j^t \\ 1 \end{pmatrix} , \quad \forall j : \mathbf{M}_{j,\nu}^t = 1 \tag{7}$$

Consequently, compositions have to be assigned to segments and the transformation matrices of the segments have to be computed. We then have to determine the matrices $\mathbf{T}_\nu^t$ and $\mathbf{M}_{j,\nu}^t$ so that the following objective function of the segmentation problem is minimized:

$$\mathcal{H}_K^t = \sum_{\nu=1}^{K} \sum_j \mathbf{M}_{j,\nu}^t \underbrace{\left\| \begin{pmatrix} \mathbf{x}_j^t \\ 1 \end{pmatrix} - \mathbf{T}_\nu^t \begin{pmatrix} \mathbf{x}_j^t \\ 1 \end{pmatrix} + \begin{pmatrix} \mathbf{h}_j^t \\ 1 \end{pmatrix} \right\|^2}_{=: \mathcal{R}\left(\mathbf{T}_\nu^t, \mathbf{h}_j^t, \mathbf{x}_j^t\right)} . \tag{8}$$

The EM-algorithm, which updates assignments and transformation matrices in alternation, is presented in Alg. 1.

Algorithm 1 EM-algorithm for computing assignments $\mathbf{M}_{j,\nu}^t$ and transformation matrices $\mathbf{T}_\nu^t$.

```
CompSegmentation($\{\mathbf{h}_j^t\}_j, \{\mathbf{T}_\nu^{t-1}\}_{\nu=1,\dots,K}$)
1 Initialization: $\forall \nu : \mathbf{T}_\nu^t \leftarrow \mathbf{T}_\nu^{t-1}$
2 repeat
3   E-Step:               ▷ update assignments:
4     $\mathbf{M}_{j,\nu}^t \leftarrow \mathbf{1}\Big\{\nu = \operatorname{argmin}_{\widehat{\nu}} \mathcal{R}\left(\mathbf{T}_{\widehat{\nu}}^t, \mathbf{h}_j^t, \mathbf{x}_j^t\right)\Big\}$
5   M-Step:               ▷ update segments:
6     for $\nu = 1, \dots, K$
7     do Solve with Levenberg-Marquardt (start with $\widehat{\mathbf{T}}_\nu^t \leftarrow \mathbf{T}_\nu^t$):
          $\mathbf{T}_\nu^t \leftarrow \operatorname{argmin}_{\widehat{\alpha},\widehat{s},\widehat{\delta}_x,\widehat{\delta}_y} \sum_j \mathbf{M}_{j,\nu}^t \mathcal{R}\left(\widehat{\mathbf{T}}_\nu^t, \mathbf{h}_j^t, \mathbf{x}_j^t\right)$
8 until  convergence of $\mathbf{M}_{j,\nu}^t$
9 return $\mathbf{M}^t$ , $\{\mathbf{T}_\nu^t\}_{\nu=1,\dots,K}$
```

One might be inclined to add an additional term into (8) that penalizes changes in the transformation matrix from one frame to the next (a momentum term). However, we observe that initializing the EM-algorithm with the solution from the previous frame (line 1 of Alg. 1) yields already stable solutions that are close to those from the previous frame (cf. [12]). Consequently, segment ν in frame t corresponds to the ν-th segment in the frame $t-1$ and we can track segments over time. In particular, the object center can be computed by

$$\mathbf{x}^{t,\nu} := \frac{1}{\sum_j \mathbf{M}_{j,\nu}^t} \sum_{j : \mathbf{M}_{j,\nu}^t = 1} \mathbf{x}_j^t . \tag{9}$$

Moreover, the initialization of the EM-algorithm leads to a convergence in less than 10 iterations on average (convergence is guaranteed since the E- and M-step minimize (8) and $\mathcal{H}_K^t$ is bounded). The $\mathbf{T}_\nu^t$ are estimated using the Levenberg-Marquardt algorithm [27, 14], which is initialized with the solution from the previous M-Step. Typically, a solution for $\mathbf{T}_\nu^t$ is found after only 3 update steps.

Using Segmentation to Refine Object Region Tracking: Alg. 2 summarizes how compositions are tracked from frame to frame by updating them with the observed interest point flows (cf. Section 2.1). Thereafter, the segmentation of the previous frame is updated using Alg. 1. Finally, in line 8 of Alg. 2, all those interest points are removed from a composition whose flow fits better to that of another object in the image. This pruning removes outlier points in compositions which occur at segment boundaries, for instance.

Algorithm 2 Tracking compositions and segmenting the frame into objects. $\bar{\mathbf{x}}_i^t$ denotes the location of interest point i in frame t, $\mathbf{x}_j^t$ is the location of composition j.

```
CompositionTracking($\{\mathbf{h}_j^{t-1}, \mathbf{x}_j^t\}_j, \{\mathbf{T}_\nu^{t-1}\}_{\nu=1,\dots,K}$)
 1 Detect interest points $i$ in frame $t$
 2 for all compositions $j$    ▷ update comps with IP flow:
 3 do $\Gamma^t(j) \leftarrow \{i : \|\mathbf{x}_j^t - \bar{\mathbf{x}}_i^t\| \leq w\}$
 4      $\mathbf{g}_j^t \leftarrow \frac{1}{|\Gamma^t(j)|} \sum_{i \in \Gamma^t(j)} \mathbf{d}_i^t$
 5      $\mathbf{h}_j^t \leftarrow \eta \mathbf{g}_j^t + (1-\eta)\mathbf{h}_j^{t-1}$
 6 $\mathbf{M}^t$ , $\{\mathbf{T}_\nu^t\}_\nu \leftarrow$ CompSegmentation($\{\mathbf{h}_j^t\}_j, \{\mathbf{T}_\nu^{t-1}\}_\nu$)
 7 for all compositions $j$   ▷ update comps with segmentat.:
 8 do $\Gamma^t(j) \leftarrow \{i : i \in \Gamma^t(j) \wedge$
              $1 = \mathbf{M}^t_{j, \operatorname{argmin}_{\widehat{\nu}} \mathcal{R}\left(\mathbf{T}_\nu^t, \mathbf{d}_i^t, \bar{\mathbf{x}}_i^t\right)}$
 9      $\mathbf{g}_j^t \leftarrow \frac{1}{|\Gamma^t(j)|} \sum_{i \in \Gamma^t(j)} \mathbf{d}_i^t$
10      $\mathbf{h}_j^t \leftarrow \eta \mathbf{g}_j^t + (1-\eta)\mathbf{h}_j^{t-1}$
11      $\mathbf{x}_j^{t+1} \leftarrow \mathbf{x}_j^t + \frac{1}{|\Gamma^t(j)|} \sum_{i \in \Gamma^t(j)} \mathbf{d}_i^t$
12 return $\{\mathbf{h}_j^t, \mathbf{x}_j^{t+1}\}_j, \{\mathbf{T}_\nu^t\}_\nu$
```

Determining the Background Segment: Finally, we assume that objects are basically forming holes in the background segment. Therefore, the segment with largest spatial extend (i.e. we compare the length of the vectors from (10)) is labeled as background. As an estimate for the height and width of the segment we use the standard deviations in x- and y-direction (the rectangular hull is by definition too sensitive w.r.t. outliers)

$$\begin{aligned}(\mathbf{b}_x^{t,\nu}, \mathbf{b}_y^{t,\nu})^\top &:= 2\lambda \cdot (\sigma_x, \sigma_y)^\top \\ &= 2\lambda \cdot \mathrm{Std}\big(\{\mathbf{x}_j^t : \mathbf{M}_{j,\nu}^t = 1\}\big)\,.\end{aligned} \tag{10}$$

$\lambda = 2$ is a reasonable choice that yields a sufficiently large covering of an object.

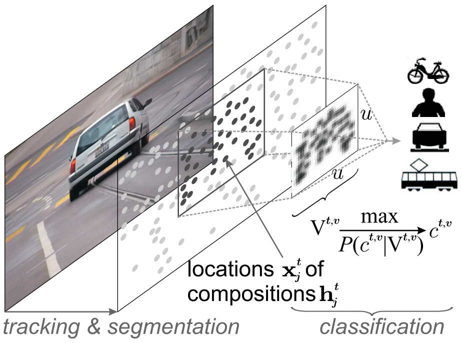
In our experiments this method for determining the background segment has worked reliably. The only noticeable failure we have observed occurred when an object was covering nearly all of a video frame and there was no background on two adjacent sides of the object, i.e. the object was zoomed in so far that it filled nearly all of the frame and was clipped at the bottom and the left. However, in the datasets we have used so far this problem was a rare exception and one could explicitly test for this case should it actually be a real problem.

3 Object Representations for Category-level Recognition

Object recognition amounts to a classification task where an object segment is classified as showing an instance of one of several object categories. The underlying object model for each category must then be learned from the training samples for that class. In the training phase compositions are tracked and objects are segmented (Alg. 2). The objects in all training frames are then represented using an object description (presented in the next sections) before training a classifier on all the samples.

3.1 Recognition using the Shape of a Dot Pattern

Motivation: Let us first investigate an object representation that is entirely based on optical flow. Now the following question arises: How can we recognize object categories based only on a set of ambiguous points (interest points with flows $\mathbf{d}_i^t$) that move through the scene when background or other objects are moving, simultaneously? Obviously, an individual point does not characterize an object class. However, the pattern of all points that are collectively moving with an object shows characteristic structure—the object shape. Therefore, an object can only be represented by a model that *jointly* describes the locations of all the points on the object.



u
u
$\mathrm{V}^{t,v} \xrightarrow[P(c^{t,v}|\mathrm{V}^{t,v})]{\max} c^{t,v}$
locations $\mathbf{x}_j^t$ of
compositions $\mathbf{h}_j^t$
tracking & segmentation
classification

Fig. 4 Object shape is represented using a grid $\mathbf{V}^{t,\nu} \in \mathbb{R}^{u \times u}$ and the object is categorized by maximizing the category posterior $P(c^{t,\nu}|\mathbf{V}^{t,\nu})$.

The approach of [21] that is based on a Hough voting of visual parts is not suited since it depends on characteristic features and does not describe dependencies between the parts. *Constellation models* [11], which describe such relationships, are only applicable to small numbers of parts for complexity reasons. Shape descriptions such as [47] establish coherent spatial mappings between a probe image and all training samples which leads to unfavorable computational costs in the recognition phase. At the other end of the modeling spectrum are bag-of-features models [7] or approaches based on *latent semantic analysis* [38] that do not represent any spatial information.

Approach: The object in segment ν is represented by laying a regular grid of $u \times u$ cells (we choose $u = 30$ to generate a sufficiently accurate representation) over the quadratic image region with diagonal length $\|\mathbf{b}^{t,\nu}\|$ around the segment center $\mathbf{x}^{t,\nu}$. Each cell indicates the distance to the nearest composition that was assigned to segment ν. The object is then represented by a matrix $\mathbf{V}^{t,\nu} \in \mathbb{R}^{u \times u}$, see Figure 4. Models similar to $\mathbf{V}^{t,\nu}$ have typically been learned from manually segmented data and commonly rely on appearance information in the grid cells, cf. [34].

The segment can then be classified as containing an object of class $c^{t,\nu}$ by maximizing the category posterior,

$$c^{t,\nu} = \operatorname*{argmax}_c P(C^{t,\nu} = c|\mathbf{V}^{t,\nu})\,. \tag{11}$$

We solve this classification problem using a multi-class SVM (RBF kernel and one-vs-one classification setting) [5]. Additionally, the dimensionality of the grid representation can be reduced by applying PCA. However, our experiments indicate that the implicit feature selection of SVMs is already sufficient so that no additional performance gain could be achieved.

3.2 Compositional, Appearance-based Model

Another approach is to describe an object with a part-based model, where each part encodes the appearance of an object region. We use the compositional model of [32] which has been applied to multi-class object categorization for more than 100 real world object categories in [32]. The following briefly summarizes the compositional approach. Local image patches at interest points are represented with appearance descriptors, i.e. the *localized feature histograms*. These are then mapped to a codebook so that compositions of interest points are represented by a histogram $\mathbf{a}_j^t$ of all the appearance descriptors that they contain. An object is then represented by coupling all the compositions based on their shift $\mathbf{S}_j^t = \mathbf{x}^{t,\nu} - \mathbf{x}_j^t$ from the object center. More precisely, compositions $\mathbf{a}_j^t$ are assumed to be conditionally independent, conditioned on the object category $c^{t,\nu}$ and object location $\mathbf{x}^{t,\nu}$. Following the derivation in [32] the category posterior can be computed by

$$P(c^{t,\nu}|\mathbf{x}^{t,\nu}, \{\mathbf{a}_j^t, \mathbf{x}_j^t\}_{j:\mathbf{M}_{j,\nu}^t=1}) \propto \prod_j \left[P(c^{t,\nu}|\mathbf{a}_j^t, \mathbf{S}_j^t = \mathbf{x}^{t,\nu} - \mathbf{x}_j^t)\right]^{\mathbf{M}_{j,\nu}^t} . \quad (12)$$

The distribution in (12) can be estimated on the training samples with a probabilistic classifier (we use a multi-class SVM as in the previous section). This model has the favorable property that it supports an arbitrary number of compositions and it is robust against missing of individual compositions (i.e. due to occlusion).

3.3 Recognition using the Motion of Dot Patterns

Now we can modify (12) to build an object model that employs the motion of object compositions, $\mathbf{h}_j^t$, relative to the motion of the object segment. Therefore, the appearance histograms $\mathbf{a}_j^t$ are substituted by $\mathbf{h}_j^t - (\mathbf{x}^{t,\nu} - \mathbf{x}^{t-1,\nu})$,

$$P(c^{t,\nu}|\mathbf{x}^{t,\nu}, \mathbf{x}^{t-1,\nu}, \{\mathbf{h}_j^t, \mathbf{x}_j^t\}_{j:\mathbf{M}_{j,\nu}^t=1}) \propto \prod_j \left[P(c^{t,\nu}|\mathbf{h}_j^t - \mathbf{x}^{t,\nu} + \mathbf{x}^{t-1,\nu},\ \mathbf{x}^{t,\nu} - \mathbf{x}_j^t)\right]^{\mathbf{M}_{j,\nu}^t} . \quad (13)$$

The individual posteriors in (13) are basically functions that map from $\mathbb{R}^4$ (flow and shift are both 2-D vectors) to a distribution over the discrete space of labels. No additional processing is required. As before, we employ an SVM for estimation.

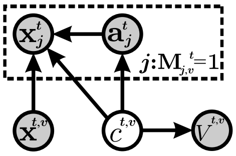


Fig. 5 Graphical model that combines global shape $\mathbf{V}^{t,\nu}$ and compositional appearance $\mathbf{a}_j^t$ at location $\mathbf{x}_j^t$ to infer the category $c^{t,\nu}$ for the object in segment ν and center $\mathbf{x}^{t,\nu}$.

3.4 Global Shape and Local Appearance Combined

Now the holistic representation for object shape in (11) and the part-based appearance model in (12) are to be combined in a single model, with the expectation that both models mutually benefit from each other. The underlying Bayesian network is presented in Figure 5. Since the $\mathbf{a}_j^t$ and the shape descriptor are conditionally independent, conditioned on $c^{t,\nu}$, we obtain for the category posterior

$$P(c^{t,\nu}|\mathbf{V}^{t,\nu}, \mathbf{x}^{t,\nu}, \{\mathbf{a}_j^t, \mathbf{x}_j^t\}_{j:\mathbf{M}_{j,\nu}^t=1}) \propto P(c^{t,\nu}|\mathbf{V}^{t,\nu}) \times \prod_j \left[P(c^{t,\nu}|\mathbf{a}_j^t, \mathbf{x}^{t,\nu} - \mathbf{x}_j^t)\right]^{\mathbf{M}_{j,\nu}^t} . \quad (14)$$

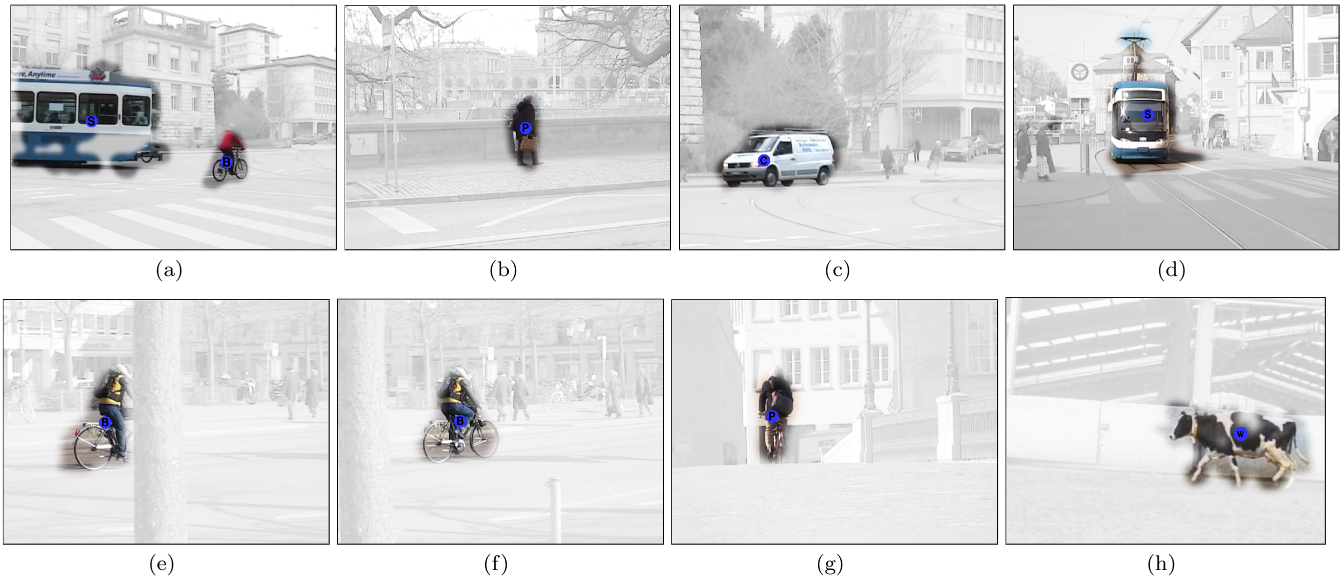
3.5 Processing Pipeline for Training

In the training phase, compositions are tracked and segmented using Alg. 2 exactly as in the recognition phase. Each video is labeled with the most prominent category it shows. Moreover, the number of segments in a frame is set to two so that only the most prominent object is found and the remaining clutter of the scene ends up in the background segment. The compositions from all foreground segments are then collected. After that, a probabilistic discriminative classifier is trained on all the gathered training data in batch mode (we use SVMs with probabilistic output [5] and RBF kernels). Depending on which object model is used, a different probability has to be estimated. For the model in (11) this means for instance that $P(C^{t,\nu} = c|\mathbf{V}^{t,\nu})$ is estimated by taking all the $\mathbf{V}^{t,\nu}$ from the training frames and using the same overall video label c for each $\mathbf{V}^{t,\nu}$ in the same video.

4 Experiments

4.1 Recognition Performance on Videos with Substantial Camera Motion

Since the presented approach automatically segments objects in cluttered videos taken by a moving camera, we evaluate it on a multi-category video database that does not support background subtraction. Therefore, we first run an experiment on the challenging database



(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 6 Segmentations and object categorizations on the dataset of [31]. The category label $c^{t,\nu}$ is placed at the location of the object center $\mathbf{x}^{t,\nu}$. (a) Simultaneous recognition of multiple objects. (e) and (f) shows object tracking despite occlusion. (g) shows a bicycle that is erroneously classified as pedestrian. (h) is a sample from the database [25] labeled as cow.

for category-level recognition in videos that has been presented in [31]. It consists of 24 videos per category (categories car, bicycle, pedestrian, and streetcar) recorded in ordinary street scenes. The videos feature large intra-class variation and the scale and viewpoint change significantly (cf. Figure 6(a), (d)) even within videos. Moreover, there is a lot of background clutter (e.g. Figure 6(b)), occlusion (see Figure 6(e)), and heavy camera motion and zooming (cf. Figure 1(b)). To make the results comparable to [31], we also use 10-fold cross-validation and train on 16 randomly chosen videos per category in each run. Testing proceeds then on the remaining videos. Object models are learned on a randomly drawn subset of 15 frames per training video, whereas testing runs over all frames of the test videos. Following common practice, retrieval rates (fraction of correctly classified test frames) are averaged per category. The overall retrieval rate ζ is, therefore, defined as

$$\zeta := \frac{1}{|\mathcal{L}|} \sum_{c \in \mathcal{L}} \{\text{true positive rate for category } c\} \ , \qquad (15)$$

where $\mathcal{L}$ denotes the set of all categories

Baseline Performance of Appearance w/o Compositions and Shape—Bag-of-Parts: The compositional approach establishes an intermediate representation that is based on compositions of parts and the spatial structure of objects. In a first experiment this hidden representation layer is neglected to evaluate the gain of compositionality. A frame is then represented using a bag-of-parts, a histogram over all the appearance features that have been extracted at all the interest points. This approach classifies $\mathbf{53.0 \pm 5.6\%}$ of all frames correctly.

Compositional Segmentation and Recognition w/o Shape Model: This experiment demonstrates the benefit of combining segmentation with recognition in a compositional model. Therefore, compositions are tracked and frames are segmented as described in Section 2. A segment is then represented using a bag--of-compositions, a histogram over the appearance descriptors $\mathbf{a}_j^t$ of all compositions in the segment. This approach filters out background using the segmentation algorithm and it utilizes compositions. However, the bag representation completely neglects the spatial layout of compositions within a segment. Consequently, only object appearance is represented but not object shape. This model yields a retrieval rate of $\mathbf{64.9 \pm 5.4\%}$ per frame.

Comparing the Different Compositional Object Models: Tab. 1 compares the retrieval rates of the different object models presented in Section 3. The flow-based representation is not suited for this dataset since these categories are primarily characterized by their shape and appearance and not by the dynamic change of articulation. In contrast to this, shape (11) provides an appropriate object representation that can compete against the approach of [31]. Moreover, this model yields significantly better performance than the two purely appearance-based bag representations that have previously been evaluated as baseline models in

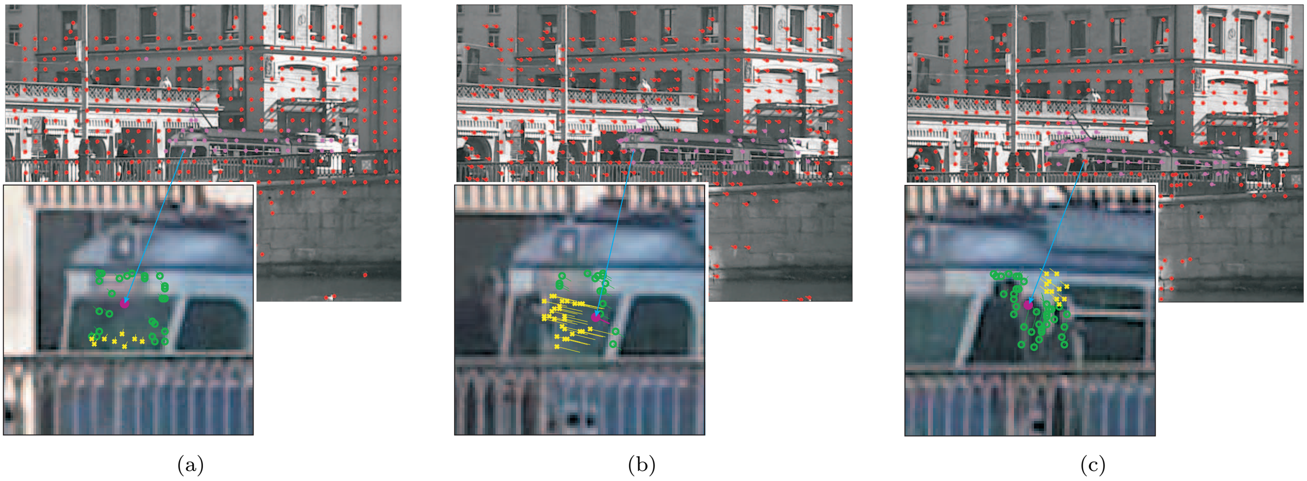


Fig. 7 Visualization of compositions. For three frames the centers $\mathbf{x}_j^t$ of compositions and their flows $\mathbf{g}_j^t$ are displayed. A segmentation with two segments is computed and visualized by coloring the compositions. The magnified regions show the same composition by visualizing the interest points assigned to the composition (circles) and those (crosses) that are rejected (using line 8 of Alg. 2) because they do not fit into the segment the composition is assigned to. In (b) the composition is partially covered by pedestrians entering from the left. In (c) the composition was dropped from the streetcar segment because it was completely covered by the pedestrians and moved with them.

this section. This result underlines that it is indeed possible to recognize objects on a category level only on the basis of moving interest points without extracting additional appearance features. Nevertheless, this representation is obviously less powerful than one that uses appearance and shape together: the model in (12) describes the appearance of compositions as well as their spatial layout within the segment. However, this model assumes that all compositions are conditionally independent, conditioned on the object category and location. Eq. (14) presents a combination of the models from (11) and (12). This combined model yields an additional performance gain on top of (12) since it models direct dependencies between the individual compositions which (12) ignores. The dependencies between compositions are captured in $P(c^{t,\nu}|\mathbf{V}^{t,\nu})$ because it jointly models the locations of all compositions. Tab. 2 presents the confusion table for the approach from (14). The most confused categories are bicycles and pedestrians. The reason for this confusion is that, when viewed from the front or back, persons riding their bike look very much like pedestrians as there is only little visual evidence for the bike (see Figure 6).

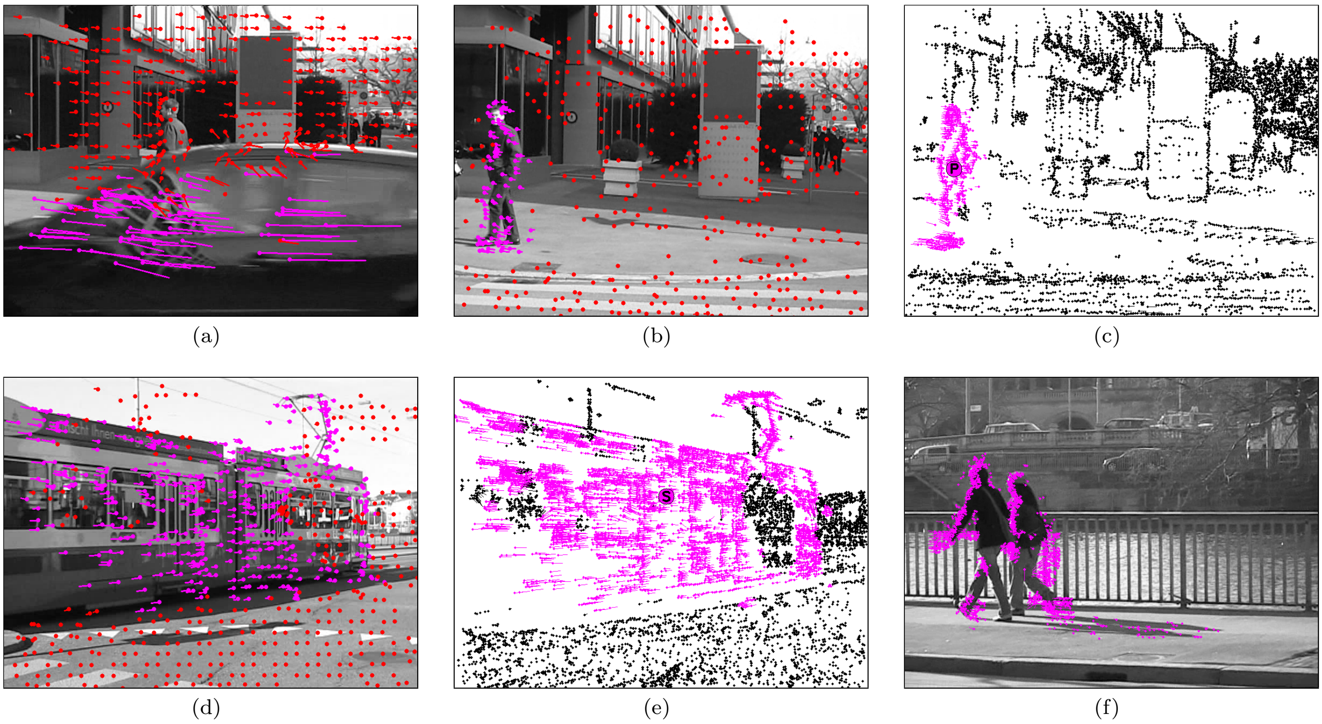
Figure 7 visualizes compositions by displaying their centers $\mathbf{x}_j^t$ and their flows $\mathbf{g}_j^t$. The segmentation (two segments) is shown by drawing the $\mathbf{x}_j^t$ in two different colors. The magnified subwindows visualize the same composition in all three frames (the disk in the middle is again $\mathbf{x}_j^t$). Therefore, all interest points within that composition are plotted—those that are actually assigned to the composition are circles, whereas the crosses show the rejected points. Interest points are rejected by line 8 of Alg. 2) when their flow fits better to another segment than to the one the composition is assigned to. In Figure 7(b) the composition is partially covered by pedestrians entering from the left so interest points on the pedestrians are rejected by the composition. Shortly after this, the composition is fully covered by the pedestrians. Therefore, it is no longer assigned to the streetcar segment but to the background

Object model	per frame	per video
Dataset of [31] (car, bicycle, pedestrian, streetcar):		
Approach of [31]	74.3 ± 4.3	87.4 ± 5.8
Compositional motion (13)	52.6 ± 1.1	68.2 ± 3.4
Appearance-only: bag-of-parts	53.0 ± 5.6	58.9 ± 6.5
Segment. w/o shape: bag-of-comps	64.9 ± 5.4	78.9 ± 5.8
Shape: $P(c^{t,\nu}\|\mathbf{V}^{t,\nu})$ (11)	74.4 ± 5.3	88.4 ± 5.2
Compositional appear + location (12)	79.6 ± 5.5	90.7 ± 5.3
Combined shape + appear (14)	81.4 ± 2.9	94.5 ± 4.9
Dataset [31] plus additional category "cow" from [25]:		
Compositional appearance (12)	76.5 ± 2.4	88.4 ± 2.3

Table 1 Retrieval rates per frame and per video (percentages) of the different object models on the dataset of [31] and on the extended dataset [31] + [25].

True classes →	bicycle	car	pedest	streetcar
bicycle	**74.3**	3.2	13.7	2.9
car	7.8	**84.1**	4.2	5.9
pedestrian	13.3	2.5	**80.0**	3.9
streetcar	4.7	10.2	2.2	**87.3**

Table 2 Category confusion table (percentages) for the combined shape and appearance model (14) per frame on the dataset [31].



(a) (b) (c)

(d) (e) (f)

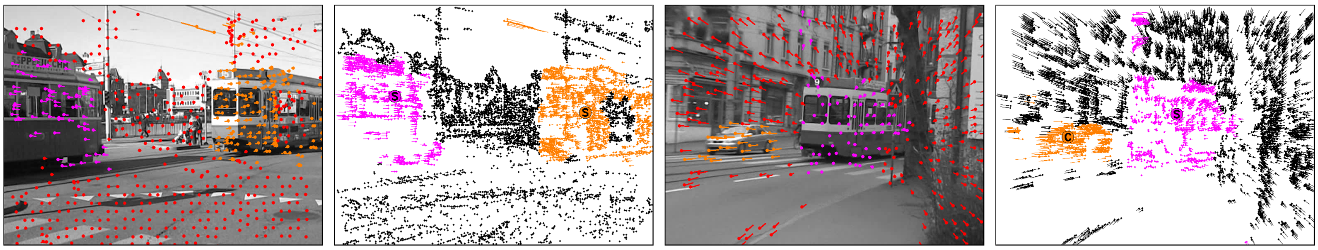
Fig. 8 (a)–(c) show recognition and segmentation under occlusion. (d) and (e) present segmentation errors due to reflection (the reflections have the same apparent motion as the background because they mirror the background). (f) shows shadow regions that end up in the object segment due to their apparent motion. Note that the flow vectors point from locations in the previous frame to locations in the current. See text for details. The figure is best viewed in color.

and starts moving with the pedestrians, see Figure 7(c). Now points on the streetcar are rejected by the composition as they do not belong to the segment the composition is assigned to. Obviously, one could reduce the impact occlusion has on compositions using a momentum term at the expense of making compositions less flexible. However, this is actually not problematic since the streetcar has now itself accumulated compositions from the background.

Figure 8 shows object recognition under occlusion and presents cases where segmentation produces incorrect results. Figure 8(a)-8(c) present a two-class segmentation for two frames of a video where a pedestrian is temporarily occluded by a passing car. While the pedestrian is covered, the segmentation switches to segment the car from the background but it resumes to segment the pedestrian once the occlusion is over. This shows that it is possible to detect objects that appear or reappear in the middle of a sequence. Whereas Figure 8(a) and 8(b) show the centers of compositions, Figure 8(c) visualizes the segmentation for the frame of Figure 8(b) by plotting all interest points and their optical flows and coloring the points according to the segmentation. Figure 8(d) shows segmented compositions and 8(e) shows the corresponding interest points. The reflections on the streetcar end up in the background segment, since their apparent motion is that of the background (the tiny area to the right of the tram is actually the mirror of the tram that is correctly put into the tram segment). A segmentation failure due to reflection can also be seen in Figure 8(a) (the car window reflects a building). Figure 8(f) shows the interest points in the foreground segment of a two-class segmentation. The shadow region moves with the pedestrians and is thus put into the same segment.

Figure 9 shows tracking, segmentation, and recognition of multiple, independently moving objects while the camera is also moving. For the same frame (a) shows the centers and flows of compositions and (b) the corresponding interest points and the color indicates the segment assignment ($K = 3$). (c) and (d) present the same visualization for another video sequence. Note the intensive camera panning and zooming in (c), (d).

We have also extended the dataset by adding the additional category *cows* (videos from [25]). Retrieval rates for model (12) are shown in Tab. 1. The relative performances of the other models w.r.t. (12) are approximately as before. It is interesting to see that although the complexity of the classification task has increased with the additional category, the performance of our



(a) (b) (c) (d)

Fig. 9 Tracking, segmentation, and recognition of multiple, independently moving objects. (a) compositions and (b) interest points for the same frame showing two streetcars. Similarly (c) and (d) show a car and a streetcar that move independently while the camera is also heavily panning and zooming (best viewed in color).

system is not significantly affected. Moreover, it underlines that our approach generalizes well to new classes and is not tuned to any particular dataset. Section 4.2 will further investigate the general applicability of the proposed approach in the completely different setting of action recognition.

Computational Demands: The system tracks, segments, and recognizes objects in videos of full PAL resolution (768×576 pixel) using the combined shape and appearance model at the order of 1 fps on a 3 GHz Pentium 4 desktop PC (videos in the standard MPEG resolution of 352×288 pixel are processed at roughly 4 fps). However, there should be still a considerable margin for optimization, e.g. by initializing the flow estimation with the solution found on the previous frame. The overall system is implemented in MatLab but all computationally demanding procedures are actually C code that is called through the MEX-interface.

Tab. 3 summarizes how the overall processing time is split up among the different subtasks. The computationally most demanding steps are interest point and flow computation as well as the computation of localized feature histograms to represent compositions $\mathbf{a}_j^t$. The EM-algorithm in Alg. 1 is efficient since it is initialized with the solution of the previous frame. Thus, the initialization leads to rapid convergence in less than 10 iterations since the algorithm is basically only updating a previous solution.

In the training phase, tracking and segmentation proceed exactly as during recognition. In this case, however, the compositions from all frames are just collected instead of using them directly to recognize objects in each frame, as in the test phase. Once all compositions have been gathered, SVM learning is conducted in batch mode on this data. This learning takes approximately 13 minutes on the database of [31] for the combined shape and appearance model of (14).

Processing step	Comp. demand
Tracking and Segmentation, Alg. 2:	
IPs i, flow $\mathbf{d}_i^t$ (Alg. 2, ln. 1)	27.7%
Updating comps (Alg. 2, ln. 2-5)	5.2%
EM estimation Alg. 1, i.e. (Alg. 2, ln. 6)	4.9%
Updating comps with segm. (Alg. 2, ln. 7-11)	0.3%
Feature extraction and Recognition:	
Computing loc feat hists to represent $\mathbf{a}_j^t$ (Sect. 3.2)	36.5%
Computing all individual probs in (14)	12.3%
Eval. GM of Fig. 5, i.e. calc. product in (14)	0.09%
Video stream ops, writing of results, etc.	12.9%

Table 3 Using the combined shape and appearance model, the approach processes full PAL video at the order of 1 fps. The table lists how much of this time is invested for the individual subprocesses.

4.2 Action Recognition

KTH Action DB: Can we turn the object categorization system into a recognition system for human action? In fact, we only have to replace the object category training samples with a database that shows different human actions. Therefore, we use the KTH human action database [35] which comprises six action types (boxing, hand clapping, hand waving, jogging, running, and walking) repeatedly performed by 25 subjects in indoor and outdoor environments. Moreover, there are also videos with scale variation and all the 600 videos (all recorded in grayscale) have a resolution of 160×120 pixels which differs significantly from the 768×576 pixel color videos in the previous experiment.

Since the task is action recognition we utilize the shape and motion based object model from Section 3.1. Appearance information would not be appropriate as it distinguishes different objects but not different actions performed by the same subject. All these videos feature one person performing an action, so we segment the person from the background by setting $K = 2$. The model from Section 3.1 does then represent actions by modeling how the individual compositions are moving with respect to the object. The confusion table for the compositional motion model (13) is presented in Tab. 4. The most confused actions are hand clapping and hand waving and there are nearly no confusions between the locomotion classes (last three) and the hand motions (first three). Combining motion with shape in the posterior $P(c^{t,\nu}|\mathbf{V}^{t,\nu}, \mathbf{x}^{t,\nu}, \mathbf{x}^{t-1,\nu}, \{\mathbf{h}_j^t, \mathbf{x}_j^t\})$ (cf. Section 3.4) does not significantly improve the performance (gain of 1%). Our investigation of this effect has shown that only small regions of the human body are actually characteristic for either of these action categories and that these distinguishing features are already captured by the motion model. Representing the complete object shape does then not yield significant extra information for discriminating the classes. The overall retrieval rate per video on all scenarios (5-fold cross-validation, training on videos for 16 persons and testing on 9 different persons) is $\mathbf{87.9 \pm 6.7}\%$ (see Tab. 5). This result significantly outperforms the 71.7% achieved by independent local features in [35]. Moreover, it turns out that the performance of our object recognition system is in the range of methods specifically designed for action recognition and which rely on background subtraction such as the HMAX approach [16], achieving retrieval rates of up to $91.6 \pm 3.0\%$.

True classes →	Box	Hclp	Hwav	Jog	Run	Walk
Boxing	**84.5**	0.0	5.5	0.0	0.0	0.0
Hand clapping	1.0	**87.0**	16.5	0.0	0.0	0.0
Hand waving	12.5	13.0	**75.5**	0.0	0.0	0.0
Jogging	0.0	0.0	0.5	**93.0**	0.0	0.0
Running	2.0	0.0	0.0	3.0	**92.3**	5.0
Walking	0.0	0.0	2.0	4.0	7.7	**95.0**

Table 4 Category confusions per video (percentages) for all scenarios of the KTH action dataset using the model of (13).

Weizmann Action DB: Finally, we also evaluate our approach on the Weizmann human action database [2] which shows 9 persons, each performing 9 different actions (running, walking, jumping-jack, jumping forward on two legs, jumping in place on two legs, galloping sideways, waving two hands, waving one hand, and bending). Subjects are roughly half as large as in the KTH set and video resolution is 180 × 144 pixel. We run 5-fold cross-validation with 6 random subjects for training and 3 for testing and summarize the results in Tab. 5 (for [16] we present the validated results for which error bars are given).

Both action recognition experiments confirm that our approach is not restricted to object categorization in videos but that it also generalizes well to other tasks such as action recognition. Moreover, the significant variation in video resolution and scene environment between the datasets [31] and [35,2] underlines that our approach does not depend on the specificities of a single recording scenario.

Approach	KTH	Weizm.
[35]	71.7	—
[29]	—	72.8
[9]	81.2	86.7 ± 7.7 (see [16])
[16]	91.6 ± 3.0	97.0 ± 3.0
Our compositional motion model (13)	87.9 ± 6.7	97.2 ± 2.5

Table 5 Recognition rates per video (percentages) on the KTH human action dataset [35] and on the Weizmann action dataset [2].

4.3 Action Recognition vs. Object Categorization

The previous experiments have demonstrated that our approach is generic and that it can be used for both action recognition and object categorization. Let us now review how both tasks are related and what their relative difficulties are. Action recognition and object categorization deal with the classification of visual patterns. Whereas objects are typically described by their shape and appearance, actions are best characterized by how their visual representation changes over time. Thus a visual representation for actions captures the change of the representation of objects (roughly speaking, actions are described by the changing of an object descriptor). Therefore, motion (13) shows much lower performance in the categorization task than shape or appearance (14). Similarly, shape and appearance are inappropriate for action recognition where motion is actually a suitable representation.

Which of the two tasks is then harder? In the specific case of our compositional model and the presented datasets, we can rank the tasks as follows: Due to its large intra-class variations, the object categorization problem on the database [31] appears to be the hardest. This problem is then followed by action recognition on the KTH dataset and finally by action recognition on the Weizmann database. In general, the difficulty of a classification task scales proportional to the intra-class variation and inverse proportional to the inter-class variation of the data—in addition there are obviously several other factors such as the degree of supervision. Consequently, both action recognition as well as object categorization can become arbitrarily difficult in the limit case and neither is per se more difficult than the other. Difficult scenarios for action recognition are for example interactions between multiple entities (e.g. "person A *helps* person B"). Similarly, functional object categories (e.g. “seating furniture”) lead to a complex categorization problem. The complexity of both of these problems arises from the fact that these classes cannot be directly characterized by their visual representation but only by a high-level, semantic concept. In the future, both tasks will require several new databases, which continue to guide research step by step towards semantic categories. Therefore, we consider it essential to increase the complexity of the problem in a controlled manner. When the intra-class variations of categories are increased, significantly more training samples, which feature the additional variability, are required.

5 Conclusion

Category-level object recognition can be accomplished by a composition machine that processes videos in the general setting of a moving camera. We have shown how recognition can be based exclusively on optical flow without requiring additional, distinctive features. Moreover, learning as well as inference do not require human intervention. Previous research has mainly focused on methods that were restricted in one of several ways (e.g. requiring that background subtraction applies, depending on manual initializations of a tracker, or being specific to only a single object class). Crucial modeling aspects are the representation and tracking based on compositions, a parametric, flow-based segmentation, and the direct link of recognition to segmentation. The approach has shown to be generic, i.e. , it is directly applicable to action recognition. In this application scenario it performs competitive to systems that are specifically designed for action recognition and for situations where background subtraction works.

References

1. S. Avidan. Ensemble tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 494–501, 2005.
2. M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1395–1402, 2005.
3. G. J. Brostow and R. Cipolla. Unsupervised bayesian detection of independent motion in crowds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 594–601, 2006.
4. G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *Proceedings of the European Conference on Computer Vision*, pages 44–57, 2008.
5. C.-C. Chang and C.-J. Lin. *LIBSVM: A library for support vector machines*, 2001.
6. D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5):564–575, 2003.
7. G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Proceedings of the European Conference on Computer Vision. Workshop Stat. Learn. in Comp. Vis.*, 2004.
8. N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *Proceedings of the European Conference on Computer Vision*, pages 428–441, 2006.
9. P. Dollar, V. Rabaud, G. Cottrell, and S. J. Belongie. Behavior recognition via sparse spatio-temporal features. In *International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 65–72, 2005.
10. P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.
11. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 264–271, 2003.
12. J. Goldberger and H. Greenspann. Context-based segmentation of image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):463–468, 2006.
13. M. Grabner, H. Grabner, and H. Bischof. Learning features for tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
14. R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.
15. Michal Irani, Benny Rousso, and Shmuel Peleg. Computing Occluding and Transparent Motions. *International Journal of Computer Vision*, 12(1):5–16, 1994.
16. H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2007.
17. Y. Jin and S. Geman. Context and hierarchy in a probabilistic image model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2145–2152, 2006.
18. M. Pawan Kumar, P. H. Torr, and A. Zisserman. Learning layered motion segmentations of video. *International Journal of Computer Vision*, 76(3):301–319, 2008.
19. S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2169–2178, 2006.
20. B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3D scene analysis from a moving vehicle. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
21. B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Proceedings of the European Conference on Computer Vision. Workshop Stat. Learn. in Comp. Vis.*, 2004.
22. V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 775–781, 2005.
23. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
24. B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

25. D. R. Magee and R. D. Boyle. Detecting lameness using 're-sampling condensation' and 'multi-stream cyclic hidden markov models'. *Image and Vision Computing*, 20(8):581–594, 2002.
26. A. Mahindroo, B. Bose, S. Chaudhury, and G. Harit. Enhanced video representation using objects. In *Proceedings of the Indian Conference on Computer Vision*, pages 105–112, 2002.
27. Donald W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.
28. G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* John Wiley, New York, NY, 1997.
29. J. C. Niebles and L. Fei Fei. A hierarchical model of shape and appearance for human action classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
30. B. Ommer and J. M. Buhmann. Learning compositional categorization models. In *Proceedings of the European Conference on Computer Vision*, pages 316–329, 2006.
31. B. Ommer and J. M. Buhmann. Compositional object recognition, segmentation, and tracking in video. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 318–333, 2007.
32. B. Ommer and J. M. Buhmann. Learning the compositional nature of visual objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
33. A. G. A. Perera, G. Brooksby, A. Hoogs, and G. Doretto. Moving object segmentation using scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Workshop on Perceptual Organization in Computer Vision*, 2006.
34. M. Pontil, S. Rogai, and A. Verri. Recognizing 3-d objects with linear support vector machines. In *Proceedings of the European Conference on Computer Vision*, pages 469–483, 1998.
35. C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *Proceedings of the International Conference on Pattern Recognition*, pages 32–36, 2004.
36. E. Seemann and B. Schiele. Cross-articulation learning for robust detection of pedestrians. In *Pattern Recognition (Symposium of the DAGM)*, pages 242–252, 2006.
37. J. Shi and C. Tomasi. Good features to track. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.
38. J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their localization in images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 370–377, 2005.
39. J. Sivic, F. Schaffalitzky, and A. Zisserman. Object level grouping for video shots. *International Journal of Computer Vision*, 67(2):189–210, 2006.
40. C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 246–252, 1999.
41. R. Vidal, Y. Ma, and S. Sastry. Generalized principal component analysis (gpca). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 621–628, 2003.
42. R. Vidal and A. Ravichandran. Optical flow estimation and segmentation of multiple moving dynamic textures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 516–521, 2005.
43. P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 734–741, 2003.
44. C. Wallraven and H. H. Bülthoff. Automatic acquisition of exemplar-based representations for recognition from image sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Workshop on Models vs. Exemplars*, 2001.
45. J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 3(5):625–638, 1994.
46. J. Y. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Proceedings of the European Conference on Computer Vision*, pages 94–106, 2006.
47. H. Zhang, A. C. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2126–2133, 2006.