Histogram-Based Description of Local Space-Time Appearance*

Karla Brkić[1], Axel Pinz[2], Siniša Šegvić[1], and Zoran Kalafatić[1]

[1] Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia
[2] Graz University of Technology, Austria

Abstract. We introduce a novel local spatio-temporal descriptor intended to model the spatio-temporal behavior of a tracked object of interest in a general manner. The basic idea of the descriptor is the accumulation of histograms of an image function value through time. The histograms are calculated over a regular grid of patches inside the bounding box of the object and normalized to represent empirical probability distributions. The number of grid patches is fixed, so the descriptor is invariant to changes in spatial scale. Depending on the temporal complexity/details at hand, we introduce "first order STA descriptors" that describe the average distribution of a chosen image function over time, and "second order STA descriptors" that model the distribution of each histogram bin over time. We discuss entropy and χ^2 as well-suited similarity and saliency measures for our descriptors. Our experimental validation ranges from the patch- to the object-level. Our results show that STA, this simple, yet powerful novel description of local space-time appearance is well-suited to machine learning and will be useful in video-analysis, including potential applications of object detection, tracking, and background modeling.

1 Introduction

Recent development of powerful detectors and descriptors has led to a tremendous boost of the success of computer vision algorithms to recognize, detect, and localize events in images. Most of these algorithms, for instance keypoint detection (DoG [11], Kadir and Brady saliency [5], MSER [13]), local scale or affine covariant description (SIFT [11], affine Harris/Laplace [14], LAF [15]), and object detection [2] are applied *at the image level*, i.e. in the 2D spatial domain. When *temporal* information (video) is available, we find that the same algorithms are still applied at a 2D image level, and the temporal aspect is often just covered by simple tracking of these 2D detections/descriptions over time.

* This research has been funded by the Croatian Science Foundation and IPV Zagreb. We also acknowledge the support by OeAD and the Croatian Ministry of Science, Education and Sports for bilateral Austrian-Croatian exchange.

A. Heyden and F. Kahl (Eds.): SCIA 2011, LNCS 6688, pp. 206–217, 2011.

A principled manner to treat the *description of local spatio-temporal events in video sequences* is still missing[1].

In this paper, we present a histogram-based descriptor for capturing the local spatio-temporal behavior of an "object" of interest. Having a description of spatio-temporal behavior at the object level opens the door for a wide variety of potential applications. Applications depend on how we view the "object" in question: is it a neighborhood of an interest point, is it a fixed rigid object with apparently moving background, such as a traffic sign seen from a moving observer, or is it a highly complex object with moving parts such as a human? Depending on the "object", we can elegantly utilize existing building blocks – for instance, a mean-shift tracker for tracking regions of interest, the Viola-Jones detector for traffic sign detection [1] or a HOG descriptor for detecting humans – to track an object of interest over time. In summary, we depart from existing 2D image-based detection and track salient events over time using existing tracking algorithms. We show a novel, principled manner to describe the local spatio-temporal behavior of objects in videos.

The benefit of having a descriptor of local spatio-temporal behavior is manyfold. At the level of interest points, consider the problem of "Multibody Structure and Motion" (MSaM [16]) analysis that requires the sparse 3D reconstruction of stationary background and a factorization of the foreground into independently moving objects. To avoid the need for many background points to be tracked, it would be very useful to identify a few, sparsely distributed "good features to track" [18] in the stationary background. At the level of fixed, rigid objects, an illustrative example comes from traffic sign detection. A traffic sign viewed from a moving car is a rigid object with a distant, moving background. But stickers that look like speed limit signs are sometimes glued to the back of a truck. A system for traffic sign detection relying solely on appearance could report such a sticker as a valid traffic sign. By modeling the local spatio-temporal behavior, however, it could be inferred that the detected object is glued to a fixed, unchanging background, so it must be a false positive. At the level of complex objects (for instance human actions, pedestrian detection and tracking), available research strongly favors the use of spatio-temporal information – be it motion trajectories, spatio-temporal volumes, or temporal HOG.

2 Related Work

The majority of work in spatio-temporal analysis concerns some type of dynamic behavior, most commonly human actions. Laptev and Perez [9] study automatic recognition of human actions in scenes taken from real movies. Their framework for detection and recognition is based on boosted window classifiers which use histogram-based spatio-temporal features. Two types of histograms are used: (i)

[1] There are a few exceptions to this observation, including the elegant extension from 2D spatial scale space theory [10] to scale in space and time [8]. But their contribution mostly covers the *detection* of local, salient space-time events at their characteristic scale, not a principled way to *describe* such events.

a HOG with four bins, to model local appearance and (ii) optical flow histograms with five bins (four orientations and one bin to represent the lack of optical flow), to model motion. Each feature is defined by the space-time cuboid on which it is calculated, by the type of the histogram used for calculation and by the mode of calculating the feature. Depending on the mode of calculation, a histogram is either calculated on the entire spatio-temporal cuboid, or the cuboid is divided into smaller parts for which individual histograms are calculated. To enable detection and recognition of actions using the proposed features, an AdaBoost classifier is trained, with Fisher Discriminants as weak learners. This classifier is combined with a purely 2D appearance classifier, which works better than any of both classifiers individually.

Ke et al. [6] focus on event detection using volumetric (i.e. spatio-temporal) features. Inspired by the success of the Viola-Jones detector, they generalize the notion of 2D rectangular features used by Viola and Jones to 3D box features. Viola and Jones themselves proposed a temporal extension of their detector intended for pedestrian detection [19], but their extension employed the differences between just two consecutive frames. The volumetric features of Ke et al., however, can span through multiple frames. The authors suggest computing the features on the optical flow of the video.

Luo et al. [12] present a learning method for human action detection in video sequences. They introduce a descriptor set named local motion histograms. Motivated by Laptev [9], they use Fisher Discriminants as weak learners on the descriptor set and then train a Gentle AdaBoost action classifier. An action is contained within a spatio-temporal volume. This volume is divided into "basic blocks" in different configurations, similar to Laptev and Perez [9]. Within each block the local motion histograms are calculated, using the magnitude and the orientation of the optical flow. Three types of histograms are defined, differing in the manner in which they are calculated (either using raw optical flow or variants of differential flow).

Dollar et al. [3] develop a framework for generic behavior detection and recognition from video sequences. Their idea is to represent a behavior by using spatio-temporal feature points, which they define as short, local video sequences such as, for instance, an eye opening or a knee bending. They propose an interest point detector intended to react to periodic motions and to spatio-temporal corners. At the interest points found by the detector they extract spatio-temporal cuboids. Each cuboid is represented by a descriptor in one of the following ways: (i) by simply flattening the cuboid into a vector, (ii) by histogramming the values in the cuboid or (iii) by dividing the cuboid into a number of regions, constructing a local histogram for each region and then concatenating all the histograms. Authors suggest histogramming either normalized pixel values, the brightness gradient, or windowed optical flow. The proposed descriptors are used in action classification by constructing a library of cuboid prototypes. A histogram of cuboid types is calculated at the level of the entire video, and is used as the behavior descriptor.

Kläser et al. [7] introduce a local descriptor for video sequences based on histograms of oriented 3D spatio-temporal gradients. The descriptor is a generalization of the well-known HOG descriptor to spatio-temporal data. The gradients become three-dimensional as they are calculated within spatio-temporal volumes using regular polyhedra. The gradient vector is positioned in the center of a regular polyhedron, and the side to which the vector points determines the histogram bin in which the vector will be placed. In their experiments, they represent video sequences as bags of words using the described spatio-temporal HOG generalization. To classify the action type, they use histograms of visual word occurences (similar to Dollar et al.) with a non-linear SVM with a χ^2 kernel.

All the approaches outlined above are intended for video analysis *once the entire video sequence is available*. In this paper, we propose a descriptor capable of harnessing spatio-temporal information on a per-frame basis, not assuming that the entire video is available. Such a descriptor can easily be used in an online setting. The descriptor is based on accumulating histograms through time. Our descriptor is not intended exclusively for action recognition – rather, it aims to model the spatio-temporal behavior of an object in a general manner.

3 Building the Spatio-temporal Appearance Descriptor

To build the spatio-temporal appearance (STA) descriptor, we require a tracked object of interest. The descriptor is calculated in every frame using the current frame information and the information from previous frames. Tracking can be achieved either by detection, or by using a standard tracker such as meanshift or KLT [17]. The algorithm for descriptor calculation assumes that a bounding box around the object of interest is available in every frame. In order to compute the descriptor, the bounding box around the object is divided into a regular grid of patches. The size of the grid is a parameter of the algorithm. For each patch, a histogram is calculated and normalized so it represents an empirical probability distribution. The value being histogrammed is a parameter of the descriptor. Possible values include hue, gradient, grayscale intensity, normalized grayscale intensity, optical flow or any other image measurement. By normalizing the histogram, i.e. representing the histogram as an empirical probability distribution, we minimize the influence of scale on the descriptor. If the histogram were absolute-valued, patches of a larger scale would have more weight. In every frame, the empirical probability distribution of each patch is updated with new measurements. The descriptor is constructed by concatenating the empirical probability distributions of all patches into a feature vector. The advantage of such an approach is that we obtain a fixed-length spatio-temporal appearance descriptor of the object in question, regardless of the spatial or temporal scale of the object. By using a grid of patches, we compensate for the possibly inaccurate object localization.

We propose two variants of the spatio-temporal appearance descriptor that differ in the level of detail in which they describe spatio-temporal behavior: (i)

spatio-temporal appearance descriptor of the first order (first-order STA descriptor), and (ii) spatio-temporal appearance descriptor of the second order (second-order STA descriptor).

3.1 Spatio-temporal Appearance Descriptor of the First Order

In the spatio-temporal appearance descriptor of the first order, each patch of the bounding box grid is represented with a single histogram, which shows the distribution of some image measurement (e.g. hue, gradient) through time.

To construct the descriptor, the bounding box around the object is in each frame divided into a regular grid of $r \times s$ patches. The n-bin histogram of the patch (u, v) is a set of bins paired with their respective relative frequencies:

$$H_{u,v} = \{(b_i, p(b_i))\}, \quad i = 1 \dots n \tag{1}$$

This histogram estimates an empirical probability distribution, where $p(b_i)$ is the *a posteriori* probability of the bin b_i. We propose integrating the histograms of an individual patch over time to obtain the first-order spatio-temporal appearance histogram (STA histogram) of the patch:

$$H_{u,v}^{(t)} = \left\{ \left(b_i, \sum_{\theta=1}^{t} \alpha_\theta p^{(\theta)}(b_i) \right) \right\} = \{(b_i, p_t(b_i))\}, \quad i = 1 \dots n \tag{2}$$

Here, we introduce the notation $p_t(b_i)$ which denotes the average empirical probability of the bin b_i in time t. The probability of bin b_i in time θ is denoted as $p^{(\theta)}(b_i)$. Parameters α_θ describe the influence of the histogram in frame θ on the overall histogram. The simplest choice for α_θ is

$$\alpha_\theta = \frac{1}{t} \tag{3}$$

which can be interpreted as histograms from all previous frames contributing equally to the final histogram. This is a good choice when we consider all the detections of the object equally valuable, regardless of *when* they were obtained. Whether all detections are considered equally valuable will depend on the nature of the problem – for instance, in the case of the observer moving towards the object, the later detections would probably be more valuable, as they would have a larger scale than the early detections. One possible way of giving more weight to the newer detections is that the integrated histogram for a given frame is equal to the average of the histogram in the current frame and the integrated histogram for all previous frames. In this case, it can be shown that the parameters α_θ are:

$$\begin{aligned} &\alpha_1 = \alpha_2 = \tfrac{1}{2^{t-1}} \\ &\alpha_\theta = \tfrac{1}{2^{t-\theta+1}} \qquad 2 < \theta \leq t \end{aligned} \tag{4}$$

assuming that the sequence has more than one frame, i.e. $t \geq 2$. The final first-order STA descriptor for an individual frame is a concatenation of the first-order STA histograms of all patches in the grid:

$$\delta^{(t)} = \left[H_{u,v}^{(t)} \right]^T, \; u = 1 \dots r, \; v = 1 \dots s \tag{5}$$

Fig. 1. Constructing the first order STA histograms for a sequence of three frames. Two patches are highlighted in red: a patch which lies in the background and a patch which lies on the object. Notice how the STA histograms of the object patch are constant through time, while the STA histograms of the background patch change.

By expanding $H^{(t)}_{u,v}$, we get:

$$\delta^{(t)} = [\underbrace{p_t(b_1)\ p_t(b_2) \dots p_t(b_n)}_{u=1,v=1}\ \underbrace{p_t(b_1)\ p_t(b_2) \dots p_t(b_n)}_{u=1,v=2}\ \dots\ \underbrace{p_t(b_1)\ p_t(b_2) \dots p_t(b_n)}_{u=r,v=s}]^T \tag{6}$$

An illustration of constructing a first order STA descriptor is shown in Fig. 1.

3.2 Spatio-temporal Histogram Descriptor of the Second Order

The first-order STA descriptor describes the distributions of some image value over a regular grid of patches through time. For simplicity, consider the behavior of the descriptor for a single patch. In the first frame, we get the distribution

of some image value for that patch. In the second frame, we get another distribution, and we update the first distribution with the new measurements so we get the integrated distribution. Therefore, in any frame our first-order descriptor will show the *average* distribution of some image value measured on the patch over time. The value of every bin of the first-order STA histogram is the average of the values of that bin in all elapsed frames (see Fig. 1). However, when considering only the average value of the bin one cannot determine how much this bin had varied through time. That information is not available in the first-order STA histogram. Therefore, we propose to model the distribution of *each histogram bin* through time. This is achieved by using histograms of second order, i.e. histograms of histograms.

The algorithm for creating a second-order STA descriptor builds on the descriptors of the first order. In every frame, the bounding box around the object is divided into a grid of $r \times s$ patches. For each patch, we calculate the patch histogram, as in (Eq. 1). Now, the bins of the obtained histograms become histogrammed themselves. The distribution of the probability $p(b_i)$ through time is modeled by a second-order STA histogram $H'^{(t)}_{u,v,i}$ with m bins β_j:

$$H'^{(t)}_{u,v,i} = \{(\beta_j, p(p_t(b_i) \in \beta_j))\}\,,\ j = 1 \dots m \tag{7}$$

This histogram describes how empirical probabilities $p_t(b_i)$ change through time. As the maximum value that $p_t(b_i)$ can take is 1, the bins β_j of the second order STA histogram will have the width of $1/m$.

The second-order STA descriptor is obtained by concatenating the second-order STA histograms into a feature vector:

$$\delta'^{(t)} = \left[H'^{(t)}_{u,v,i}\right]^T,\ u = 1 \dots r,\ v = 1 \dots s \tag{8}$$

As explained in Subsection 3.1, the first-order STA descriptor describes the average appearance of an object through time. In contrast, the second-order descriptor encodes both the object appearance and the change of that appearance.

4 Learning from the STA Descriptor

Having built a spatio-temporal appearance descriptor, it is interesting to review possible saliency measures which can be applied to the descriptor to distinguish different kinds of space-time behavior. Both variants of the STA descriptor $\delta^{(t)}$ are a concatenation of histogram probabilities. We simplify the notation and denote every element of the histogram descriptor d_k. Hence, the STA descriptor of the first order is:

$$\delta^{(t)} = [d_1\ d_2 \dots\ d_k]^T,\quad k = r \times s \times n \tag{9}$$

while the STA descriptor of the second order is:

$$\delta'^{(t)} = [d_1\ d_2 \dots\ d_k]^T,\quad k = r \times s \times n \times m \tag{10}$$

4.1 Entropy

Because every element of our descriptor originates in a histogram and estimates a probability, we can calculate the total entropy of the descriptor by:

$$E(\delta^{(t)}) = -\sum_k d_k \log d_k \tag{11}$$

which is essentially the sum of entropies of histograms which were concatenated into the descriptor[2]. The formula is valid for first and second order descriptors.

Entropy of the STA descriptor conveys important information about the behavior of the object through time. Consider the case of the first-order STA descriptor. If a patch changes a lot through time, its first-order STA histogram will approach a uniform distribution – because if the patches were changing completely randomly, every bin of the histogram would be equally likely. On the other hand, if the patch remains fairly constant through time, we expect a stable and constant histogram. As entropy is a measure of randomness, a larger entropy will indicate a distribution closer to uniform. Therefore, using entropy, we can distinguish between patches that vary and patches that stay the same. There is, however, one problem: by measuring the entropy of the first-order STA histogram, we cannot distinguish between a patch which is constant through time, but has an appearance resulting in a uniform histogram, and a patch whose appearance varies a lot through time. Both cases lead to a uniform first-order STA histogram. To address this, one can measure the total entropy of the second-order STA descriptor. As the STA histogram of the second order models the *change* in the first-order STA histogram, the entropy we obtain will be invariant to the object appearance.

We envision two uses for the entropy measure. First, at the level of a single object, knowing the parameters of the descriptor and having a training set of descriptors $\delta^{(t)}$ one can find which patches inside the grid of the object bounding box are temporally stable – i.e., which patches are likely to describe the object, and which patches are likely to describe the background. Second, at the level of multiple objects, one can compare total entropies of two different objects to find which object is more stable through time. This has proved to be especially useful in finding good features to track (see the experimental section).

4.2 The χ^2 Measure

The spatio-temporal behavior of an object can also be investigated using the χ^2 measure. This measure shows whether some empirical probability distribution matches with the theoretically expected distribution. In a general experiment, the χ^2 measure is calculated as

$$\chi^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{12}$$

[2] We denote entropy by E, because H is already in use for histograms.

with O_i being the observed frequency and E_i being the expected frequency. In the context of our histograms, we can use the χ^2 measure to determine how much a patch changes through time (similarly to the entropy measure). Suppose that we wish to determine whether a patch changes a lot. If it were changing a lot, we would expect its first-order STA histogram to be fairly uniform. Hence, we choose a null hypothesis that the part of the descriptor corresponding to the histogram of one patch represents a uniform distribution.

Mathematically, assume that the descriptor is given by Eq. 6. For patch $u = 1,\ v = 1$, the observed values are $p_t(b_i),\ i = 1 \ldots n$, while the expected values correspond to a uniform distribution and thus are $\mu(p_t(b_i)) = 1/n$. Then, the χ^2 measure of similarity of the patch (u, v) with a uniform distribution is:

$$\chi^2_{u,v} = \sum_{i=1}^{n} \frac{(p_t(b_i) - 1/n)^2}{p_t(b_i)} \tag{13}$$

Using this measure, we can determine the similarity of the observed distribution with a uniform distribution, which might provide an important clue to whether a patch is changing or not.

4.3 Using the STA Descriptor in Machine Learning

The STA descriptor can be used directly as a feature vector in any machine learning algorithm. The descriptor length is a constant, regardless of the number of frames through which the object spans or the scale of the object. At the same time, the descriptor is richer in information than a single image of an object, because it includes the temporal dimension as well. Instead of using the descriptor directly, one can first transform it by applying one of the mentioned saliency measures on the elements of the descriptor which correspond to STA histograms of individual patches. In case of the first-order STA descriptor this means applying the saliency measures on the histograms of patch appearance, while in case of the second-order STA this means applying them on the histograms of such histograms. Using the descriptor as a feature vector, we can train a classifier that discriminates between various classes of objects. Depending on the desired level of complexity, we will use either the first-order or the second-order descriptor. The training set is constructed by tracking the objects through time and calculating the descriptors in frames of interest. Depending on the application, one might choose to calculate the descriptor of the object in every frame, and thus obtain more training samples, or to calculate the descriptor in several selected frames, or perhaps just in the last frame. An important constraint to keep in mind is the dimensionality of the descriptor, which can be quite large, especially for the second-order descriptor (if we assume a grid of 5×5 patches, and $m = n = 5$, then the dimensionality of the second order descriptor will be $r \times s \times m \times n = 5^4 = 625$). In order to train a classifier which uses such a descriptor, one needs a large number of training samples. Possible classifiers which might be suitable include neural networks, support vector machines, k-NN classifiers, tree-based classifiers, variants of boosting etc.

5 Illustrative Experiments

To present the benefits of using the STA descriptor, we chose three illustrative examples: discriminating between true and false positives, discriminating between static and dynamic background and finding good features to track.

5.1 Discriminating between True and False Positives

Object detectors, when applied to large amounts of data such as videos, inevitably produce false positive detections. To deal with that, one usually trains additional classifiers exploiting different classification cues. Here, we analyze the benefit of training one such classifier on STA descriptors of the object over training it on object images without the temporal component. Our positive samples are triangular traffic signs tracked using a combination of the Viola-Jones detector and the KLT tracker. For negatives, we choose two variants: (i) artificial false positives – background patches which are randomly selected and then tracked and (ii) real false positives obtained as the responses of the Viola-Jones detector trained on traffic signs [1]. To build the training set, we calculate the first-order STA descriptor of the object in every frame, and add the descriptor to the set with the corresponding label (object / non-object). Hence, for every frame in which the object appears we obtain one training sample. In calculating the descriptor, we use a grid of 5×5 patches and 10 histogram bins. The value being histogrammed is hue. For the classifier, we use a random forest of 10 trees. When using real false positives, the total number of training samples is 17806, while the total number of testing samples is 1978. When using artificial false positives, the total number of training samples is 25370, while the number of testing samples is 2818. Results summarized in Table 1 show that by using the first-order STA descriptor we reduce the number of false positives and obtain much better ROC curves than when working with raw data.

Table 1. Results of discriminating objects (traffic signs) and non-objects (false positives) using different types of false positives (artificial examples or examples obtained by the Viola-Jones detector), different operators (hue, gradient) and different feature vectors (raw pixels / HOG vs first-order STA). The employed classifier is a random forest. We show true positive (TP), false positive (FP), true negative (TN) and false negative (FN) rates for the decision threshold of .5.

negatives	function	feature vector	TP	FN	FP	TN	AuROC
artificial	hue	raw pixels	0.994	0.006	0.172	0.828	0.903
artificial	hue	first-order STA	0.981	0.019	0.018	0.982	0.989
Viola-Jones	hue	raw pixels	0.843	0.157	0.168	0.832	0.898
Viola-Jones	hue	first-order STA	0.840	0.160	0.101	0.899	0.947
Viola-Jones	gradient	raw HoG	0.851	0.149	0.336	0.664	0.831
Viola-Jones	gradient	first-order STA	0.868	0.132	0.080	0.920	0.960

5.2 Distinguishing between a Static and a Moving Background

Using the proposed saliency measures and the second-order STA descriptor, we can train a classifier which distinguishes between objects of the same class that are glued to a static background and objects which have a moving, distant background. To illustrate this fact, we created an artificial training set consisting of tracked triangular signs on a static background and tracked triangular signs with a moving background. An image of a sign is first selected from a database of 2000 real traffic sign images and masked to remove its background. Then the artificial background is randomly selected from a set of available backgrounds. We simulate the tracking of the sign through time by enlarging the sign and the background by a plausible random value until the sign reaches some predefined scale limit. For the class of signs with the moving background, we also simulate background motion. Additionally, we simulate localization noise by randomly offsetting the bounding box around the sign. In every frame, we create the second-order STA descriptor of the object. The descriptor is calculated over a grid of 5×5 patches and 10 histogram bins are used both for the first-order and the second-order histogram. The value being histogrammed is gradient orientation. We calculate the entropy of each second-order histogram and form a feature vector by concatenating all the calculated entropies. The dimensionality of the feature vector is then equal to the dimensionality of the first-order STA descriptor: 250. We use around 40000 training samples and around 10000 testing samples. To allow motion to develop, we include only the descriptors of the frames after frame 3 of the object. The trained random forest classifier achieves a true positive rate of 0.999 and a false positive rate of 0.125, which shows that the proposed descriptor successfully models change.

5.3 Finding Stable Features to Track

Finally, we collected first experimental evidence regarding the benefit of our novel STA descriptors for the problem of finding good features to track in the background of complex Multibody Structure and Motion (MSaM) scenes. We analyzed a recent MSaM secquence by Holzer and Pinz [4], where their original algorithm detects and tracks about 200 point features in the scene. Typically, 150-180 of these points are located in stationary background. We harvested the most salient background features by ordering all the points by the entropy of their first-order STA descriptors in every frame and selecting the top 20 points. These points can be seen as a sparse reconstruction of the stationary background and can be used in terms of "good features to track" [18] the camera pose.

6 Conclusion and Outlook

The main contribution of this paper certainly is a fundamental one: we have introduced STA - a novel spatio-temporal appearance descriptor based on histograms. We believe that STA will be widely used and highly successful in many

applications of video processing due to its simplicity and general applicability. The descriptor combines spatial and temporal information into a fixed-length feature vector, independent of spatial or temporal scale of an object. Our proposed saliency measures are helpful in analyzing the space-time behavior of the object further. We have illustrated how the descriptor can be applied in different use cases, from discriminating between objects to finding good features to track.

In our future work, we plan to use STA descriptors for the analysis of complex Multibody Structure and Motion (MSaM) scenes, and for the learning and discrimination of category specific motion patterns.

References

1. Brkić, K., Pinz, A., Šegvić, S.: Traffic sign detection as a component of an automated traffic infrastructure inventory system. In: Proc. 33rd ÖAGM Workshop, Stainz, Austria (May 2009)
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR (2005)
3. Dollár, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: VS-PETS (October 2005)
4. Holzer, P., Pinz, A.: Mobile surveillance by 3d-outlier analysis. In: Proc. ACCV Workshop on Visual Surveillance (2010)
5. Kadir, T., Brady, M.: Scale, saliency and image description. Int. J. Computer Vision 45(2), 83–105 (2001)
6. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: Proc. ICCV, vol. 1, pp. 166–173 (October 2005)
7. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: British Machine Vision Conference, pp. 995–1004 (September 2008)
8. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proc. ICCV (2003)
9. Laptev, I., Perez, P.: Retrieving actions in movies. In: Proc. ICCV, pp. 1–8 (2007)
10. Lindeberg, T.: Scale Space theory in Computer Vision. Kluwer, Dordrecht (1994)
11. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. J. Computer Vision (2), 91–110 (2004)
12. Luo, Q., Kong, X., Zeng, G., Fan, J.: Human action detection via boosted local motion histograms. Mach. Vision Appl. 21, 377–389 (2010)
13. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In: Proc. 13th BMVC, pp. 384–393 (2002)
14. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis & Machine Intelligence 27(10), 1615–1630 (2005)
15. Obdržálek, S., Matas, J.: Object recognition using local affine frames on distinghuished regions. In: Proc. 13th BMVC, pp. 113–122 (2002)
16. Ozden, K., Schindler, K., van Gool, L.: Multibody structure-from-motion in practice. IEEE PAMI 32(6), 1134–1141 (2010)
17. Šegvić, S., Remazeilles, A., Chaumette, F.: Enhancing the point feature tracker by adaptive modelling of the feature support. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 112–124. Springer, Heidelberg (2006)
18. Shi, J., Tomasi, C.: Good features to track. In: Proc. CVPR, pp. 593–600 (1994)
19. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. Int. J. Computer Vision 63, 153–161 (2005)