

Evaluation of local spatio-temporal features for action recognition

Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, Cordelia Schmid

▶ **To cite this version:**

Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. BMVC 2009 - British Machine Vision Conference, Sep 2009, London, United Kingdom. pp.124.1-124.11, 10.5244/C.23.124 . inria-00439769

HAL Id: inria-00439769
https://inria.hal.science/inria-00439769

Submitted on 8 Apr 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Evaluation of local spatio-temporal features for action recognition

Heng Wang[1,3]
hwang@nlpr.ia.ac.cn
Muhammad Muneeb Ullah[2]
Muhammad.Muneeb.Ullah@inria.fr
Alexander Kläser[1]
Alexander.Klaser@inria.fr
Ivan Laptev[2]
Ivan.Laptev@inria.fr
Cordelia Schmid[1]
Cordelia.Schmid@inria.fr

[1] LEAR, INRIA, LJK
Grenoble, France
[2] VISTA, INRIA
Rennes, France
[3] LIAMA, NLPR, CASIA
Beijing, China

Abstract

Local space-time features have recently become a popular video representation for action recognition. Several methods for feature localization and description have been proposed in the literature and promising recognition results were demonstrated for a number of action classes. The comparison of existing methods, however, is often limited given the different experimental settings used. The purpose of this paper is to evaluate and compare previously proposed space-time features in a common experimental setup. In particular, we consider four different feature detectors and six local feature descriptors and use a standard bag-of-features SVM approach for action recognition. We investigate the performance of these methods on a total of 25 action classes distributed over three datasets with varying difficulty. Among interesting conclusions, we demonstrate that regular sampling of space-time features consistently outperforms all tested space-time interest point detectors for human actions in realistic settings. We also demonstrate a consistent ranking for the majority of methods over different datasets and discuss their advantages and limitations.

1 Introduction

Local image and video features have been shown successful for many recognition tasks such as object and scene recognition [8, 17] as well as human action recognition [16, 24]. Local space-time features capture characteristic shape and motion in video and provide relatively independent representation of events with respect to their spatio-temporal shifts and scales as well as background clutter and multiple motions in the scene. Such features are usually extracted directly from video and therefore avoid possible failures of other pre-processing methods such as motion segmentation and tracking.

Many different space-time feature detectors [6, 10, 14, 22, 26, 27] and descriptors [12, 15, 16, 25, 26] have been proposed in the past few years. Feature detectors usually select

© 2009. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

spatio-temporal locations and scales in video by maximizing specific saliency functions. The detectors usually differ in the type and the sparsity of selected points. Feature descriptors capture shape and motion in the neighborhoods of selected points using image measurements such as spatial or spatio-temporal image gradients and optical flow.

While specific properties of detectors and descriptors have been advocated in the literature, their justification is often insufficient due to the limited and non-comparable experimental evaluations used in current papers. For example, results are frequently presented for different datasets such as the KTH dataset [6, 10, 12, 16, 24, 26, 27], the Weizmann dataset [3, 25] or the aerobic actions dataset [22]. For the common KTH dataset [24], results are often non-comparable due to the different experimental settings used. Furthermore, most of the previous evaluations were reported for actions in controlled environments such as in KTH and Weizmann datasets. It is therefore unclear how these methods generalize to action recognition in realistic setups [16, 23].

Several evaluations of local space-time features have been reported in the past. Laptev [13] evaluated the repeatability of space-time interest points as well as the associated accuracy of action recognition under changes in spatial and temporal video resolution as well as under camera motion. Similarly, Willems et al. [26] evaluated repeatability of detected features under scale changes, in-plane rotations, video compression and camera motion. Local space-time descriptors were evaluated in Laptev et al. [15], where the comparison included families of higher-order derivatives (local jets), image gradients and optical flow. Dollár et al. [6] compared local descriptors in terms of image brightness, gradient and optical flow. Scovanner et al. [25] evaluated the 3D-SIFT descriptor and its two-dimensional variants. Jhuang et al. [10] evaluated local descriptors in terms of the magnitude and orientation of space-time gradients as well as optical flow. Kläser et al. [12] compared space-time HOG descriptor with HOG and HOF descriptors [16]. Willems et al. [26] evaluated the extended SURF descriptor. However, evaluations in these works were usually limited to a single detection or description method as well as to a single dataset.

The current paper overcomes above-mentioned limitations and provides a fair comparison for a number of local space-time detectors and descriptors. We evaluate performance of three space-time interest point detectors and six descriptors along with their combinations on three datasets with varying degree of difficulty. Moreover, we compare with dense features obtained by regular sampling of local space-time patches, as recently excellent results were obtained by dense sampling in the context of object recognition [7, 11]. We, furthermore, investigate the influence of spatial video resolution and shot boundaries on the performance. We also compare methods in terms of their sparsity as well as the speed of available implementations. All experiments are reported for the same bag-of-features SVM recognition framework. Among interesting conclusions, we demonstrate that regular sampling consistently outperforms all tested space-time detectors for human actions in realistic setups. We also demonstrate a consistent ranking for the majority of methods across datasets.

The rest of the paper is organized as follows. In Section 2, we give a detailed presentation of the local spatio-temporal features included in our comparison. Section 3 then presents the experimental setup, i.e., the datasets and the bag-of-features approach used to evaluate the results. Finally, Section 4 compares results obtained for different features and Section 5 concludes the paper with the discussion.

2 Local spatio-temporal video features

This section describes local feature detectors and descriptors used in the following evaluation. Methods were selected based on their use in the literature as well as the availability of the implementation. In all cases we use the original implementation and parameter settings provided by the authors.

2.1 Detectors

The **Harris3D** detector was proposed by Laptev and Lindeberg in [14], as a space-time extension of the Harris detector [9]. The authors compute a spatio-temporal second-moment matrix at each video point $\mu(\cdot;\sigma,\tau) = g(\cdot;s\sigma,s\tau) * (\nabla L(\cdot;\sigma,\tau)(\nabla L(\cdot;\sigma,\tau))^T)$ using independent spatial and temporal scale values σ,τ, a separable Gaussian smoothing function g, and space-time gradients ∇L. The final locations of space-time interest points are given by local maxima of $H = \det(\mu) - k\,\mathrm{trace}^3(\mu)$, $H > 0$. The authors proposed an optional mechanism for spatio-temporal scale selection. This is not used in our experiments, but we use points extracted at multiple scales based on a regular sampling of the scale parameters σ,τ. This has shown to give promising results in [16]. We use the original implementation available on-line[1] and standard parameter settings $k = 0.0005$, $\sigma^2 = 4,8,16,32,64,128$, $\tau^2 = 2,4$.

The **Cuboid** detector is based on temporal Gabor filters and was proposed by Dollár *et al.* [6]. The response function has the form: $R = (I * g * h_{ev})^2 + (I * g * h_{od})^2$, where $g(x,y;\sigma)$ is the 2D spatial Gaussian smoothing kernel, and h_{ev} and h_{od} are a quadrature pair of 1D Gabor filters which are applied temporally. The Gabor filters are defined by $h_{ev}(t;\tau,\omega) = -\cos(2\pi t\omega)e^{-t^2/\tau^2}$ and $h_{od}(t;\tau,\omega) = -\sin(2\pi t\omega)e^{-t^2/\tau^2}$ with $\omega = 4/\tau$. The two parameters σ and τ of the response function R correspond roughly to the spatial and temporal scale of the detector. Interest points are the local maxima of the response function R. We use the code from the authors' website[2] and detect features using standard scale values $\sigma = 2, \tau = 4$.

The **Hessian** detector was proposed by Willems *et al.* [26] as a spatio-temporal extension of the Hessian saliency measure used in [2, 18] for blob detection in images. The detector measures the saliency with the determinant of the 3D Hessian matrix. The position and scale of the interest points are simultaneously localized without any iterative procedure. In order to speed up the detector, the authors used approximative box-filter operations on an integral video structure. Each octave is divided into 5 scales, with a ratio between subsequent scales in the range $1.2-1.5$ for the inner 3 scales. The determinant of the Hessian is computed over several octaves of both the spatial and temporal scales. A non-maximum suppression algorithm selects joint extrema over space, time and scales: (x,y,t,σ,τ). We use the executables from the authors' website[3] and employ the default parameter setting.

Dense sampling extracts video blocks at regular positions and scales in space and time. There are 5 dimensions to sample from: (x, y, t, σ, τ), where σ and τ are the spatial and temporal scale, respectively. In our experiments, the minimum size of a 3D patch is 18×18 pixels and 10 frames. (In Section 4.4, we evaluate different spatial patch sizes for dense sampling.) Spatial and temporal sampling are done with 50% overlap. Multi-scale patches

[1] http://www.irisa.fr/vista/Equipe/People/Laptev/download.html#stip
[2] http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html
[3] http://homes.psat.kuleuven.be/~gwillems/research/Hes-STIP/

are obtained by multiplying σ and τ by a factor of $\sqrt{2}$ for consecutive scales. In total, we use 8 spatial and 2 temporal scales since we consider the spatial scale to be more important than the time scale. We consider all combinations of spatial and temporal scales, i.e., we sample a video 16 times with different σ and τ parameters.

2.2 Descriptors

For each given sample point (x,y,t,σ,τ), a feature descriptor is computed for a 3D video patch centered at (x,y,t). Its spatial size $\Delta_x(\sigma),\Delta_y(\sigma)$ is a function of σ and its temporal length $\Delta_t(\tau)$ a function of τ. Dollár et al. [6] proposed the **Cuboid** descriptor along with the Cuboid detector. The size for the descriptor is given with $\Delta_x(\sigma) = \Delta_y(\sigma) = 2 \cdot \text{ceil}(3\sigma) + 1$ and $\Delta_t(\tau) = 2 \cdot \text{ceil}(3\tau) + 1$. We follow the authors' setup and concatenate the gradients computed for each pixel in the patch into a single vector. Then, principal component analysis (PCA) is used to project the feature vector to a lower dimensional space. We download the code from the authors' website[2] and use the default settings (*e.g.*, the size of descriptor after PCA projection is 100). The PCA basis is computed on the training samples.

The **HOG/HOF** descriptors were introduced by Laptev et al. in [16]. To characterize local motion and appearance, the authors compute histograms of spatial gradient and optic flow accumulated in space-time neighborhoods of detected interest points. For the combination of HOG/HOF descriptors with interest point detectors, the descriptor size is defined by $\Delta_x(\sigma) = \Delta_y(\sigma) = 18\sigma, \Delta_t(\tau) = 8\tau$. Each volume is subdivided into a $n_x \times n_y \times n_t$ grid of cells; for each cell, 4-bin histograms of gradient orientations (*HOG*) and 5-bin histograms of optic flow (*HOF*) are computed. Normalized histograms are concatenated into HOG, HOF as well as HOG/HOF descriptor vectors and are similar in spirit to the well known SIFT descriptor. In our evaluation we used the grid parameters $n_x, n_y = 3$, $n_t = 2$ as suggested by the authors. We noticed low dependency of results for different choices of the scale factor for σ,τ in general. We use the original implementation available on-line[1]. When computing HOG/HOF descriptors for Hessian detectors, we optimized the mappings $\sigma = \alpha\sigma^h$ and $\tau = \beta\tau^h$ w.r.t. α,β for HOG/HOF scale parameters σ,τ and the scale parameters σ^h,τ^h returned by the Hessian detector. For the Cuboid detector (computed at low space-time scale values) we fixed the scales of HOG/HOF descriptors to $\sigma^2 = 4$ and $\tau^2 = 2$.

The **HOG3D** descriptor was proposed by Kläser et al. [12]. It is based on histograms of 3D gradient orientations and can be seen as an extension of the popular SIFT descriptor [20] to video sequences. Gradients are computed using an integral video representation. Regular polyhedrons are used to uniformly quantize the orientation of spatio-temporal gradients. The descriptor, therefore, combines shape and motion information at the same time. A given 3D patch is divided into $n_x \times n_y \times n_t$ cells. The corresponding descriptor concatenates gradient histograms of all cells and is then normalized. We use the executable from the authors' website[4] and apply their recommended parametric settings for all feature detectors: descriptor size $\Delta_x(\sigma) = \Delta_y(\sigma) = 8\sigma, \Delta_t(\tau) = 6\tau$, number of spatial and temporal cells $n_x = n_y = 4, n_t = 3$, and icosahedron as polyhedron type for quantizing orientations.

Willems *et al.* [26] proposed the **extended SURF (ESURF)** descriptor which extends the image SURF descriptor [1] to videos. Like for previous descriptors, the authors divide 3D patches into $n_x \times n_y \times n_t$ cells, The size of the 3D patch is given by $\Delta_x(\sigma) = \Delta_y(\sigma) = 3\sigma, \Delta_t(\tau) = 3\tau$. For the feature descriptor, each cell is represented by a vector of weighted

[4] http://lear.inrialpes.fr/software



Figure 1: Sample frames from video sequences of KTH (top), UCF Sports (middle), and Hollywood2 (bottom) human action datasets.

sums $v = (\sum d_x, \sum d_y, \sum d_t)$ of uniformly sampled responses of the Haar-wavelets d_x, d_y, d_t along the three axes. We use the executables from the authors' website[3] with the default parameters defined in the executable.

3 Experimental setup

In this section we describe the datasets used for the evaluation as well as the evaluation protocol. We evaluate the features in a bag-of-features based action classification task and employ the evaluation measures proposed by the authors of the datasets.

3.1 Datasets

We carry out our experiments on three different action datasets which we obtained from the authors' websites. The **KTH actions** dataset [24][5] consists of six human action classes: walking, jogging, running, boxing, waving, and clapping (cf. Figure 1, top). Each action class is performed several times by 25 subjects. The sequences were recorded in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. The background is homogeneous and static in most sequences. In total, the data consists of 2391 video samples. We follow the original experimental setup of the authors, i.e., divide the samples into test set (9 subjects: 2, 3, 5, 6, 7, 8, 9, 10, and 22) and training set (the remaining 16 subjects). As in the initial paper [24], we train and evaluate a multi-class classifier and report average accuracy over all classes as performance measure.

The **UCF sport actions** dataset [23][6] contains ten different types of human actions: swinging (on the pommel horse and on the floor), diving, kicking (a ball), weight-lifting, horse-riding, running, skateboarding, swinging (at the high bar), golf swinging and walking (cf. Figure 1, middle). The dataset consists of 150 video samples which show a large intra-class variability. To increase the amount of data samples, we extend the dataset by adding

[5] Available at http://www.nada.kth.se/cvap/actions/
[6] Available at http://www.cs.ucf.edu/vision/public_html/

a horizontally flipped version of each sequence to the dataset. Similar to the KTH actions dataset, we train a multi-class classifier and report the average accuracy over all classes. We use a leave-one-out setup and test on each original sequence while training on all other sequences together with their flipped versions (i.e., the flipped version of the tested sequence is removed from the training set).

The **Hollywood2 actions** dataset [21][7] has been collected from 69 different Hollywood movies. There are 12 action classes: answering the phone, driving car, eating, fighting, getting out of the car, hand shaking, hugging, kissing, running, sitting down, sitting up, and standing up (see Figure 1, bottom). In our experiments, we used the clean training dataset (the authors also provide an automatic, noisy dataset). In total, there are 1707 action samples divided into a training set (823 sequences) and a test set (884 sequences). Train and test sequences are obtained from different movies. The performance is evaluated as suggested in [21], i.e., by computing the average precision (AP) for each of the action classes and reporting the mean AP over all classes (mAP).

3.2 Evaluation framework

A video sequence is represented as a bag of local spatio-temporal features [24]. Spatio-temporal features are first quantized into visual words and a video is then represented as the frequency histogram over the visual words. In our experiments, vocabularies are constructed with k-means clustering. We set the number of visual words V to 4000 which has shown to empirically give good results for a wide range of datasets. To limit the complexity, we cluster a subset of 100,000 randomly selected training features. To increase precision, we initialize k-means 8 times and keep the result with the lowest error. Features are assigned to their closest vocabulary word using Euclidean distance. The resulting histograms of visual word occurrences are used as video sequence representations.

For classification, we use a non-linear support vector machine [5] with a χ^2-kernel [16]

$$K(H_i,H_j) = exp(-\frac{1}{2A}\sum_{n=1}^{V}\frac{(h_{in}-h_{jn})^2}{h_{in}+h_{jn}}) \ , \tag{1}$$

where $H_i = \{h_{in}\}$ and $H_j = \{h_{jn}\}$ are the frequency histograms of word occurrences and V is the vocabulary size. A is the mean value of distances between all training samples [28]. For multi-class classification, we apply the *one-against-rest* approach and select the class with the highest score.

4 Experimental results

This section presents experimental results for various detector/descriptor combinations. Results are presented for the different datasets in Sections 4.1-4.3. Section 4.4 evaluates different parameters for dense sampling. The computational complexity of the tested methods is evaluated in Section 4.5

Due to high memory requirements of some descriptor/detector code, we subsample original UCF and Hollywood2 sequences to half spatial resolution in all our experiments. This enables us to compare all methods on the same data. We evaluate the effect of subsampling for the Hollywood2 data set in Section 4.3. The ESURF and Cuboid descriptors are not

[7] Available at http://www.irisa.fr/vista/Equipe/People/Laptev/download.html

	HOG3D	HOG/HOF	HOG	HOF	Cuboids	ESURF
Harris3D	89.0%	**91.8%**	80.9%	**92.1%**	–	–
Cuboids	**90.0%**	88.7%	82.3%	88.2%	89.1%	–
Hessian	84.6%	88.7%	77.7%	88.6%	–	81.4%
Dense	85.3%	86.1%	79.0%	88.0%	–	–

Table 1: Average accuracy for various detector/descriptor combinations on the KTH dataset.

	HOG3D	HOG/HOF	HOG	HOF	Cuboids	ESURF
Harris3D	79.7%	78.1%	71.4%	75.4%	–	–
Cuboids	**82.9%**	77.7%	72.7%	76.7%	76.6%	–
Hessian	79.0%	79.3%	66.0%	75.3%	–	77.3%
Dense	**85.6%**	81.6%	77.4%	**82.6%**	–	–

Table 2: Average accuracy for various detector/descriptor combinations on the UCF dataset.

evaluated for other detectors than those used in original papers. Unfortunately, separate implementations of these descriptors were not available. Note that due to random initialization of k-means used for codebook generation, we observed a standard deviation of approximately 0.5% in our experiments.

4.1 KTH actions dataset

KTH actions [24][5] is to date the most common dataset in evaluations of action recognition. Among recently reported results, Laptev *et al*. [16] obtain 91.8% using a combination of HOG and HOF descriptors, while Kläser *et al*. [12] get 91.4% with the HOG3D descriptor. Both methods use the Harris3D detector and follow the original experimental setup of [24]. Adopting the Cuboid detector, Liu and Shah [19] report 94.2%, and Bregonzio *et al*. [4] obtain 93.2% with a 2D Gabor filter based detector. Note, however, that these results were obtained for a simpler Leave-One-Out Cross-Validation setting and are not directly comparable to results in this paper.

Our results for different combinations of detectors and descriptors evaluated on KTH are presented in Table 1. The best results are obtained for Harris3D + HOF (92.1%) and HOG/HOF (91.8%). These results are comparable to 91.8% reported in [16] for Harris3D + HOG/HOF. For Harris3D + HOG3D, we only reach 89.0%, about 2.5% lower than the original result in [12]. This could be explained by the different strategy of codebook generation (random sampling) used in [12]. For the Cuboid detector, the best result 90.0% is obtained with the HOG3D descriptor. The performance of Hessian and Dense detectors are below Harris3D and Cuboids. The low performance of dense sampling on KTH may be explained by the large number of features corresponding to the static background. When comparing performance of different descriptors, we note that HOG/HOF and HOF give best results in combination with Harris3D, Hessian and Dense features.

4.2 UCF sports dataset

The results for different combinations of detectors and descriptors evaluated on **UCF sport actions** are illustrated in Table 2. The best result 85.6% over different detectors is obtained by the dense sampling. This can be explained by the fact that dense features capture different types of motions. Furthermore, they also capture background which may provide useful context information. Scene context indeed may be helpful for sports actions which often involve specific equipment and scene types as illustrated in Figure 1. The second-best result

	HOG3D	HOG/HOF	HOG	HOF	Cuboids	ESURF
Harris3D	43.7%	45.2%	32.8%	43.3%	–	–
Cuboids	45.7%	**46.2%**	39.4%	42.9%	45.0%	–
Hessian	41.3%	**46.0%**	36.2%	43.0%	–	38.2%
Dense	45.3%	**47.4%**	39.4%	45.5%	–	–

Table 3: Mean AP for various detector/descriptor combinations on the Hollywood2 dataset.

	HOG3D	HOG/HOF	HOG	HOF
reference	43.7%	45.2%	32.8%	43.3%
w/o shot boundary features	43.6%	**45.7%**	35.3%	43.4%
full resolution videos	45.8%	**47.6%**	39.7%	43.9%

Table 4: Comparison of the Harris3D dector on (top) videos with half spatial resolution, (middle) with removed shot boundary features, and (bottom) on the full resolution videos.

82.9% is obtained for the Cuboid detector. Also above 80% are dense points in combination with HOG/HOF and HOF. Harris3D and Hessian detectors perform similar at the level of 80%. Among different descriptors, HOG3D provides best results for all detectors except Hessian. HOG/HOF gives second-best result for UCF. The authors of the original paper [23] report 69.2% for UCF. Their result, however, does not correspond to the version of UCF dataset available on-line[6] used in our evaluation.

4.3 Hollywood2 dataset

Finally, evaluation results for **Hollywood2 actions** are presented in Table 3. As for the UCF dataset, the best result 47.4% is obtained for dense sampling while interest point detectors demonstrate similar and slightly lower performance. We assume dense sampling again benefits from a more complete description of motions and the rich context information. Among different descriptors, HOG/HOF performs best. Unlike in results for **KTH actions**, here the combination of HOF and HOG improves HOF with about 2 percent. The HOG3D descriptor performs similar to HOF.

Shot boundary features. Since action samples in Hollywood2 are collected from movies, they contain many shot boundaries, which cause many artificial interest points. To investigate the influence of shot boundaries on recognition results, we compare in Table 4 the performance of the Harris3D detector with and without shot boundary features. Results for HOG/HOF and HOG demonstrate 0.5% and 2% improvement respectively when removing shot boundary features while the change in performance for other descriptors is minor. We conclude that shot boundary features do not influence our evaluation significantly.

Influence of subsampling. We also investigate the influence of reduced spatial resolution adopted in our Hollywood2 experiments. In Table 4 recognition results are reported for videos with full and half spatial resolution using the Harris3D detector. The performance is consistently and significantly increased for all tested descriptors for the case of full spatial resolution. Note that for full resolution, we obtain approximately 3 times more features per sequence than for half resolution.

4.4 Dense sampling parameters

Given the best results obtained with dense sampling, we further investigate the performance as a function of different minimal spatial sizes of dense descriptors (cf. Table 5). As before, further spatial scales are sampled with a scale factor of $\sqrt{2}$. As in Sections 4.2 and 4.3, we

Spatial Size	Hollywood2				UCF			
	HOG3D	HOG/HOF	HOG	HOF	HOG3D	HOG/HOF	HOG	HOF
18×18	**45.3%**	47.4%	39.4%	45.5%	**85.6%**	**81.6%**	**77.4%**	82.6%
24×24	45.1%	**47.7%**	**39.4%**	**45.8%**	82.0%	81.4%	76.8%	**84.0%**
36×36	44.8%	47.3%	36.8%	45.6%	78.6%	79.1%	76.5%	82.4%
48×48	42.8%	46.5%	35.8%	45.5%	78.8%	78.6%	73.9%	79.0%
72×72	39.7%	45.2%	32.2%	43.0%	77.8%	78.8%	69.6%	78.4%

Table 5: Average accuracy for dense sampling with varying minimal spatial sizes on the Hollywood2 and UCF sports dataset.

	Harris3D + HOG/HOF	Hessian + ESURF	Cuboid Detector + Descriptor	Dense + HOG3D	Dense + HOG/HOF
Frames/second	1.6	4.6	0.9	0.8	1.2
Features/frame	31	19	44	643	643

Table 6: Average speed and average number of generated features for different methods.

present results for Hollywood2 and UCF videos with half spatial resolution. We observed no significant improvements for different temporal lengths, therefore we fixed the temporal length to 10 frames. The overlapping rate for dense patches is set to 50%. We can see that the performance increases with smaller spatial size, i.e., when we sample denser. However, the performance saturates in general at a spatial size of 24×24 for Hollywood2 and 18×18 for UCF.

4.5 Computational complexity

Here we compare the tested detectors by their speed and the number of detected interest points. The comparison was performed on a set of videos from Hollywood2 with spatial resolution of 360×288 pixels (half resolution) and about 8000 frames length in total. The run-time estimates were obtained on a Dell Precision T3400 Dual core PC with 2.66 GHz processors and 4GB RAM. Table 6 presents results for the three detectors and dense sampling in terms of average number of frames per second and average number of features per frame. Note that feature computation is included in the run time. Among the detectors, Cuboid extracts the densest features (44 features/frame) and it is the slowest one (0.9 frames/second). Hessian extracts the sparsest features (19 features/frame) and is consequently the most efficient (4.6 frames/second). As for the dense sampling, since there was no feature detection as such, the overall computational time was mainly spent on the feature description. Obviously, dense sampling extracts many more features than interest point detectors. Note that the time of descriptor quantization was not taken into account in this evaluation.

5 Conclusion

Among the main conclusions, we note that dense sampling consistently outperforms all tested interest point detectors in realistic video settings, but performs worse on the simple KTH dataset. This indicates both (a) the importance of using realistic experimental video data as well as (b) the limitations of current interest point detectors. Note, however, that dense sampling also produces a very large number of features (usually 15-20 times more than feature detectors). This is more difficult to handle than the relatively sparse number of interest points. We also note a rather similar performance of interest point detectors for

each dataset. Across datasets, Harris 3D performs better on KTH dataset, while the Cuboid detector gives better results for UCF and Hollywood2 datasets.

Among the tested descriptors, the combination of gradient based and optical flow based descriptors seems to be a good choice. The combination of dense sampling with the HOG/HOF descriptor provides best results for the most challenging Hollywood2 dataset. On the UCF dataset, the HOG3D descriptor performs best in combination with dense sampling. This motivates further investigations of optical flow based descriptors.

Acknowledgements. This work was partially funded by the European research project CLASS, the MSR/INRIA joint project, and the QUAERO project.

References

[1] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *ECCV*, 2006.

[2] P. Beaudet. Rotationally invariant image operators. In *ICPR*, 1978.

[3] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[4] M. Bregonzio, S. Gong, and T. Xiang. Recognising action as clouds of space-time interest points. In *CVPR*, 2009.

[5] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001.

[6] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005.

[7] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *CVPR*, 2005.

[8] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR*, 2003.

[9] C. Harris and M.J. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1988.

[10] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. In *ICCV*, 2007.

[11] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *ICCV*, 2005.

[12] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3D-gradients. In *BMVC*, 2008.

[13] I. Laptev. *Local Spatio-Temporal Image Features for Motion Interpretation*. PhD thesis, Department of Numerical Analysis and Computer Science (NADA), KTH, 2004.

[14] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003.

[15] I. Laptev and T. Lindeberg. Local descriptors for spatio-temporal recognition. In *First International Workshop on Spatial Coherence for Visual Motion Analysis*, LNCS. Springer, 2004.

[16] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[17] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[18] T. Lindeberg. Feature detection with automatic scale selection. *IJCV*, 30(2):79–116, 1998.

[19] J. Liu and M. Shah. Learning human actions via information maximization. In *CVPR*, 2008.

[20] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[21] M. Marszałek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[22] A. Oikonomopoulos, I. Patras, and M. Pantic. Spatio-temporal salient points for visual recognition of human actions. *IEEE Trans. Systems, Man, and Cybernetics, Part B*, 36 (3):710–719, 2006.

[23] M. Rodriguez, J. Ahmed, and M. Shah. Action mach: A spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[24] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *ICPR*, 2004.

[25] P. Scovanner, S. Ali, and M. Shah. A 3-dimensional SIFT descriptor and its application to action recognition. In *ACM International Conference on Multimedia*, 2007.

[26] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *ECCV*, 2008.

[27] S.F. Wong and R. Cipolla. Extracting spatio-temporal interest points using global information. In *ICCV*, 2007.

[28] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2): 213–238, 2007.