Exemplar-based Action Recognition in Video

Geert Willems[1]
homes.esat.kuleuven.be/~gwillems
Jan Hendrik Becker[1]
homes.esat.kuleuven.be/~jhbecker
Tinne Tuytelaars[1]
homes.esat.kuleuven.be/~tuytelaa
Luc Van Gool[12]

[1] ESAT-PSI/Visics
K.U. Leuven
[2] Computer Vision Laboratory
BIWI/ETH Zürich

Abstract

In this work, we present a method for action localization and recognition using an exemplar-based approach. It starts from local dense yet scale-invariant spatio-temporal features. The most discriminative visual words are selected and used to cast bounding box hypotheses, which are then verified and further grouped into the final detections. To the best of our knowledge, we are the first to extend the exemplar-based approach using local features into the spatio-temporal domain. This allows us to avoid the problems that typically plague sliding window-based approaches - in particular the exhaustive search over spatial coordinates, time, and spatial as well as temporal scales. We report state-of-the-art results on challenging datasets, extracted from real movies, for both classification and localization.

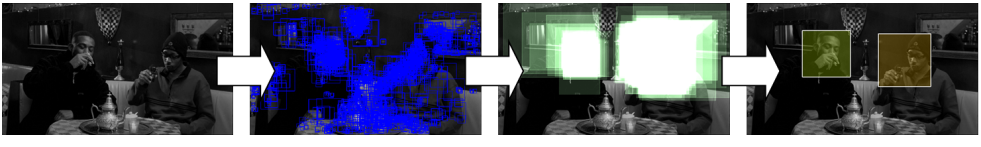


Figure 1: An illustration of the detection pipeline for the 'drinking' action: (1) a frame from the input video, (2) spatio-temporal features are computed throughout the video, (3) exemplar-based hypotheses are generated and scored, (4) a grouping algorithm yields two distinct detections.

1 Introduction

With the amount of user-generated video content on the web growing dramatically (*e.g.* 65,000 new video clips uploaded on YouTube™ on a daily basis), the need for automatic annotation or content-based retrieval of video data is, arguably, even larger than is the case of still images. Indeed, manual annotations, if provided at all, are often short, inaccurate, incomplete and subjective. Moreover, they typically lack precise time stamps and hence do not allow to fast forward to the relevant part of a video.

BMVC 2009 doi:10.5244/C.23.90

Over recent years, a lot of progress has been made towards automatic annotation of video material, especially in the context of object and scene recognition (mostly working on keyframes extracted from the video [13]). However, in comparison, action recognition is still in its infancy. Yet, for indexing and retrieval purposes, it is important to know not only who (or which objects) are present in the scene, but even more so what it is they are doing.

Related work Whereas originally silhouette-based approaches (*e.g.* [1, 7, 20]) or approaches based on pose estimation (*e.g.* [4, 15, 19]) have been studied mostly, good results have been reported recently using extensions of traditional object recognition approaches to the spatio-temporal domain [5, 12, 16, 22]. These methods are not limited to human actions. Instead, they consider actions as typical spatio-temporal patterns that can be modeled using local features, optical flow, or gradient-based descriptors. It is in this line of research that our work is situated. More specifically, we build on the work of Chum *et al.* [3], which is an exemplar-based approach for object detection using local features.

Roughly speaking, methods for object detection (localization) can be divided into sliding window-based approaches and constellation-based approaches, where the former are clearly predominant (see *e.g.* recent editions of the Pascal VOC challenge [6]). Straightforward extensions of these techniques to the spatio-temporal domain are not very practical though. Sliding window-based approaches exhaustively scan the image with windows of increasing scale. This typically results in a 3-dimensional search space. However, for actions (spatio-temporal patterns), this exhaustive search would become prohibitively expensive, as both the spatial location, the temporal location, as well as the spatial and temporal scale dimensions need to be scanned, resulting in a 5-dimensional search space. If the aspect ratio is not fixed, as in our experiments, even 6 dimensions need to be searched. For some actions, like walking or running, non-rectangular bounding boxes such as parallellepipeds, with even more free parameters, might be a better choice. Laptev and Perez [12] circumvent this problem by using a cascade, where the first stage uses only a single frame (detecting the keyframe of the action). Ignoring the motion-information in this first step seems suboptimal though. Moreover, not all actions have a clearly defined keyframe (*e.g.* for 'HugPerson' this is not so clearly defined as for 'Drinking'). Lampert *et al.* [10] have proposed a branch-and-bound scheme to avoid the exhaustive search, and this approach could easily be extended to the spatio-temporal domain. However, this is only feasible for a particular class of classifiers (*e.g.* linear SVM's only).

Apart from sliding window-based approaches, constellation-based approaches have been popular for object detection as well. A good example in this category is the Implicit Shape Model (ISM) [14]. But again, an extension to the spatio-temporal domain is not straightforward, as this would result in a 5- or 6-dimensional Hough space where votes for possible spatio-temporal locations and scales are accumulated. Moreover, this method cannot easily cope with dense sets of features, whereas generally it has been shown that dense feature sets yield better performance than sparse sets.

An exemplar-based approach Recently, a different approach based on object exemplars has been proposed by Chum *et al.* [3], that is especially appealing because of its simplicity. This method can be situated somewhere in between sliding window-based approaches and the ISM model. Chum *et al.* focus on the problem of weakly supervised object detection, where no bounding boxes are available during training. Instead, bounding boxes are generated iteratively in an optimization procedure initialized by discriminative local features.

Detection then follows a similar hypothesize-and-test framework, where bounding boxes are hypothesized based on the position and scale of discriminative local features and evaluated based on a spatial pyramid based classifier (but any other classifier could be used as well). This approach effectively avoids the exhaustive search, only evaluating those windows for which at least one consistent discriminative feature has been found. Especially in the case of action recognition, where the exhaustive search of sliding window-based approaches results in a serious overhead, this approach seems promising and worth exploring. This is the core contribution of this paper. To keep the results clean and easily interpretable, we kept the system relatively simple (*e.g.* no spatial binning of the bounding box, no clustering of exemplars). Even then, we obtain state-of-the-art results. It is important to emphasize that the 'exemplar-based' in the title has to be interpreted as in Chum *et al.*'s paper, in that bounding boxes are 'copied' from examples seen during training, followed by a standard appearance-based classification. It should not be confused with action recognition methods that try to match 2-dimensional silhouettes (*e.g.* [7, 20]) or 3-dimensional silhouette induced space-shapes [8] from a test video with an (often static) key pose, which are sometimes also referred to as 'exemplar-based'.

A second contribution of this paper consists of the evaluation of different local spatio-temporal features and descriptors in the context of action recognition, using the challenging Hollywood movies action dataset of [13]. Additionally, we present an extended version of the 'DrinkingSmoking' dataset [12]. The remainder of this paper is organized as follows. First, we discuss the local feature-based image representation (section 2). Next, we introduce our pipeline for exemplar-based action recognition from video (section 3). In section 4, we describe the datasets used for classification and localization and evaluate the results. We end with an overall discussion and conclusion.

2 Video representation

In all our experiments, we use a very simple video representation, based on a bag-of-words computed over an entire video clip (for classification) or over a given space-time volume (for detection). The overall results can probably still be improved further by switching to more complex representations, such as spatial pyramids. However, we want to avoid too much parameter finetuning and believe that, for clarity, a simple representation is better suited to showcase the intrinsic quality of different descriptors and/or detection approaches.

Spatio-temporal local features We use the dense, scale-invariant, spatio-temporal *Hes-STIP* detector of Willems *et al.* [21]. This detector responds to spatio-temporal blobs within a video, based on an approximation of the determinant of the Hessian. As shown in [21], those features have the advantage that they are scale-invariant (both spatially as well as temporally), yet are not so sparse as other scale-invariant spatio-temporal feature detectors such as [11]. This may be advantageous in our setting, as in the context of object recognition it has been shown that denser feature sets are typically to be preferred. Also the scale invariance is important, as in our exemplar-based approach the scale of the hypothesized bounding boxes directly depends on the scale of the features. A multiscale approach, as used in [13], might thus be not well suited in this setting.

Spatio-temporal descriptor In [9], Kläser *et al.* propose a new variant of a local histogram-of-gradients spatio-temporal descriptor, where the orientation of the sampled gradients is

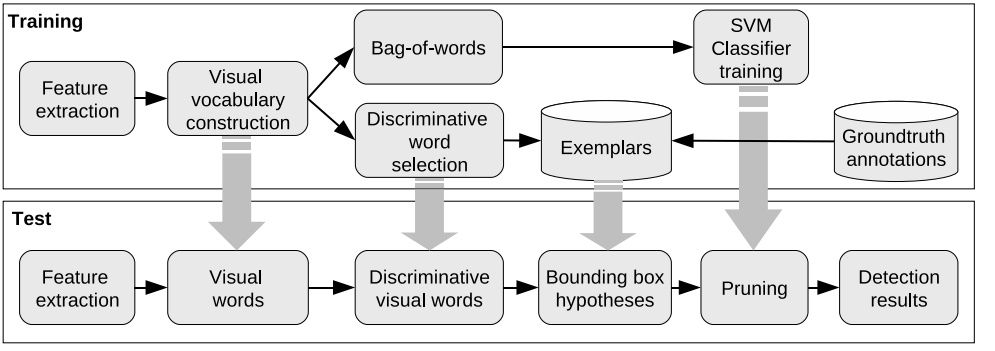


Figure 2: Overview of the exemplar-based detection pipeline.

first quantized before it is added to a histogram. The quantized orientations correspond to the normals of the faces of a platonic solid, of which there exist only five: a tetrahedron, a cube, an octahedron, a dodecahedron and an icosahedron. This results in respectively 4, 6, 8, 12, and 20 quantized orientations. Willems *et al*. [21] also proposed a novel spatio-temporal feature descriptor recently, inspired by the SURF descriptor [2], which we will denote by *stSURF*. Similar to SURF, the gradients are computed along the 3 main axes and both the sum of the gradients as well as the sum of the absolute values of the gradients are stored. This is comparable to a quantization using the 6-sided cube. Based on the improvement shown in [9], we extend the *stSURF* descriptor with orientation quantization. An important difference between the 3D gradient descriptors of [9] and *stSURF* lies in the way the gradients within a cuboid are computed. While the former averages precomputed gradients within this volume, the latter computes the gradient at the scale of the volume using three axis-aligned Haar-wavelets. We refer to the new descriptor using 20 quantized directions as *stSURF20*.

3 Exemplar-based action recognition

Next, we discuss the actual exemplar-based action detection scheme. Figure 1 illustrates the main steps of the detection pipeline on a fragment of the test video and figure 2 gives a detailed overview of the system consisting of a training and a detection pipeline.

3.1 Training pipeline

For training, we start with a set of videos where each instance of the action we want to detect is annotated (we use a simple 3-dimensional bounding box, but the proposed method can easily deal with more complex annotation regions as well). Within each of these spatio-temporal annotation regions (boxes), local spatio-temporal features (section 2) are extracted and vector quantized using approximate k-means [17]. Next, we select the top N most discriminative visual words (see below). We collect exemplars for each selected word, by searching for all features belonging to that word and storing the coordinates of the annotation region relative to the feature's position and scale, as in [3]. Unlike Chum *et al*. [3], we do not cluster the bounding boxes, but instead keep all of them (similar to what is done in the ISM model). At the same time, we also learn a classifier, based on the ground truth annotations in the training data as well as a set of randomly generated negatives (not overlapping with any of the ground truth bounding boxes). We compute a bag-of-words for each annotation and train

a non-linear support vector machine using the χ^2-kernel. This kernel has previously been reported to yield very good classification results [9].

Selecting discriminative words In order to select the top N most discriminative words for each action, we use a F_β-measure-based ranking. The F_β-measure is defined as the harmonic mean of precision and recall, with recall weighted β-times as much as precision. We found a value of $\beta = 0.01$ to work best, as we rather want to select words that occur mainly in the relevant samples (i.e. have high precision) and not words that occur too often in the non-relevant samples. Selecting the top $N = 50$ words from that ranked word list ensures that 90% of the training actions contain at least one of the discriminative words.

3.2 Detection pipeline

Given a test video, we start the detection pipeline by extracting all local spatio-temporal features within the video. Each extracted feature that belongs to one of the top N most discriminative visual words generates a set of hypotheses. Next, a pruning step removes hypotheses that are not deemed useful because of their bounding box properties or due to a low confidence level. In a final step, we group the remaining hypotheses. Each group of hypotheses is called a detection. A confidence value is given to each detection based on the hypotheses belonging to the group.

Generating hypotheses Each extracted feature is assigned to its visual word using the visual vocabulary obtained in the training stage. If this word belongs to the top N most discriminative words, one hypothesis is created from each exemplar of that word. The hypothesized bounding boxes are positioned based on the location and scales of the feature.

Pruning hypotheses The initial set of obtained hypotheses can be quite large. However, many of them have unrealistic dimensions or positions. We therefore, in a first step, remove all hypotheses that do not fall entirely inside the video (with a small leniency) or have too long a duration. For the hypotheses that are kept, we compute the bag-of-words from the features inside their bounding box and assign as a confidence the decision value of the previously trained SVM classifier. In a second step, we further prune all hypotheses with a confidence value below a pre-defined threshold.

Generating detections Because the exemplars are not clustered during training as in [9], we typically end up (even after pruning) with many overlapping hypotheses that should be grouped before computing the final ranking. For this, we use a simple, greedy algorithm. In the list of ungrouped hypotheses, we search for the hypothesis with the highest confidence value (largest distance to the SVM decision hyperplane) and use it as a seed for a detection. Next, we add to this detection all ungrouped hypotheses that overlap with the seed hypothesis. This grouping step is repeated until all hypotheses belong to a detection. The bounding box of a detection is defined by the average center position and dimensions of the bounding boxes of its hypotheses. Finally, a confidence measure is given to each detection. We have tested four different measures: the number of hypotheses belonging to the detection, the confidence of the seed hypothesis, the average hypothesis confidence and the sum of hypothesis confidences. Out of these, none came out a clear winner in all cases, yet the confidence of the seed hypothesis scored well overall. This confidence measure is also used in the experiments in section 4.2.

Dealing with shot cuts Movie segments, like the test video, typically contain many shot cuts. As the grouping algorithm just looks for overlapping hypotheses, it is possible that hypotheses on both sides of a shot cut are merged into one detection. Since we only want to group hypotheses that were generated from features belonging to one distinct action, we look at the position of the feature that created the seed hypothesis. Other overlapping hypotheses will only be added if their feature is located between the same two shot cuts.

4 Experimental evaluation

We report on both classification and localization results. The main purpose of the classification experiments is to evaluate our choice of local features and descriptors, as well as to indicate the relative difficulty of all three datasets used. The detection experiments then evaluate our exemplar-based approach. But we start with a description of the datasets.

Datasets We use three different datasets to evaluate our work. The *Hollywood Human Actions* dataset [13] contains clips of eight actions taken from 32 different Hollywood movies. Typically there is only one action per clip, though in some cases a clip may have several labels, when different actions happen at the same time. It is a very challenging dataset as it combines vastly different appearances of actions in a more or less real-world setting. At the same time the amount of training data is limited (cf. [13]), which makes it hard for recognition systems to generalize well beyond the training samples. Unfortunately, the location of actions is not well defined, rendering the dataset difficult to use for the detection task. In some instances the real action is not even shown within the clip, but must rather be deduced from the context and position of people visible, like two people shaking hands while merely their upper bodies are shown, excluding the hands. For all our experiments we use only the clean training and testing set as described in [13].
The *DrinkingSmoking* (DS) dataset [12] contains 160 drinking and 149 smoking samples (of which 38 and 14 belong to the test set respectively) taken from two Hollywood movies as well as some extra recordings. For each action instance a bounding box is provided that localizes the action within the video. We generated 273 negative training and 47 negative testing bounding boxes by random selection of space-time volumes within those movies, which are similar in size and aspect-ratio to positive annotations but without overlapping any of them. We further created an *ExtendedDrinkingSmoking* (EDS) dataset by adding three more movies[1] to the training part, which now contains 288 drinking and 304 smoking samples in total. In the same manner as before, we increased the number of negative training samples to 529 while keeping the original test set. The annotations of the extended dataset are available online[2].

Parameter settings In all our experiments, unless mentioned otherwise, we use feature extraction with a threshold of 0.001 on the determinant of the Hessian and a descriptor as discussed in section 2 with $4 \times 4 \times 3$ subdivisions, each containing a histogram over 20 orientations. For a feature with spatial scale σ and temporal scale τ, we compute the descriptor inside a support region of 9σ and 10τ, respectively. The detector spans 3 spatial and 2 temporal octaves. The visual vocabulary contains 4096 words and the SVM model is trained using a χ^2-kernel with 5-fold cross-validation for parameter optimization.

[1] The following three movies were added to the dataset: "Carlito's Way" (48 drinking, 38 smoking), "Scent of a Woman" (22 drinking, 9 smoking), and "Scarface" (58 drinking, 108 smoking)

[2] http://homes.esat.kuleuven.be/~gwillems/ExamplarBasedActions

	Laptev +HoG [13]	Laptev +HoF [13]	Laptev +Kläser [9]	HesSTIP +stSURF20	HesSTIP +Kläser	HesSTIP + stSURF20b
AnswerPhone	13.4%	24.6%	18.6% (±1.9)	25.2% (±5.4)	21.2% (±2.9)	22.9% (±4.4)
HugPerson	29.1%	17.4%	19.8% (±1.1)	20.5% (±5.9)	17.2% (±5.3)	19.5% (±5.6)
GetOutCar	21.9%	14.9%	22.6% (±2.1)	23.3% (±4.7)	17.6% (±4.6)	20.4% (±4.7)
HandShake	18.6%	12.1%	11.8% (±1.3)	15.3% (±2.9)	17.6% (±3.5)	17.9% (±6.0)
SitDown	29.1%	20.7%	32.5% (±7.2)	30.4% (±3.2)	37.4% (±2.3)	33.8% (±1.9)
SitUp	6.5%	5.7%	7.0% (±0.6)	17.3% (±6.1)	14.7% (±7.8)	21.8% (±6.4)
Kiss	52.0%	36.5%	47.0% (±0.7)	49.7% (±2.7)	50.5% (±2.8)	50.2% (±2.1)
StandUp	45.4%	40.0%	38.0% (±1.3)	48.7% (±1.2)	48.8% (±2.4)	49.8% (±2.5)
Average	27.0%	21.5%	24.7% (±2.0)	28.8% (±4.0)	28.1% (±4.0)	29.6% (±4.2)

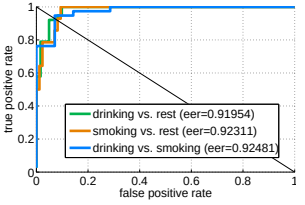
Table 1: Classification results on Hollywood dataset in terms of average precision. The first three columns use multiscale features, the last three scale-invariant *HesSTIPs*. The last three columns also show mean and standard deviation over 10 runs.

4.1 Classification results

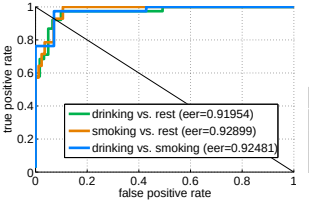
For classification, we use a bag-of-words representation of all action annotations (positive and negative) with a vocabulary obtained via approximate k-means clustering [17] on the training samples and aforementioned non-linear support vector machines. We show our results in terms of average precision (AP) which is often used in retrieval contexts. In order to achieve a high AP score, the system must retrieve (correctly classify) all the relevant documents as early as possible. Raw SVM decision-values are used to rank the classifications.

Hollywood actions We use the Hollywood actions dataset to compare the performance of the local spatio-temporal feature detector and descriptor used in this paper, *HesSTIP* [21] + *stSURF20* (cf. section 2), with some alternative detector-descriptor schemes reported in the literature in the context of action recognition. We consider the multiscale features from Laptev *et al*. (*Laptev* [13]) together with the following descriptors: standard histogram-of-gradients (*HoG* [13]), histogram-of-optical-flow (*HoF* [13]), and 3D gradient descriptor with 20 orientations (*Kläser* [9]). Additionally we run an experiment where we apply the default support range of the Kläser descriptor to the newly proposed *stSURF20* descriptor. We refer to this instance of the descriptor as *stSURF20b*. Due to the random initialisation of the clustering employed for feature quantization, we report the mean average precisions and their standard deviation over 10 runs. Detailed results are given in table 4.1, where features are named using *detector+descriptor* convention. We use the default settings of the binaries, provided by the authors, to compute the multiscale Harris-Laplace and the scale-invariant Hessian-based space-time features. We use the binary provided by Kläser [9] to describe both types of interest points. Multiscale detectors typically generate many more features than scale-invariant detectors. In our case the number of extracted multiscale features (5.3M) is more than twice the amount of the scale-invariant features (1.9M). Comparison of *HesSTIP+Kläser* and *Laptev+Kläser* shows, however, a clear performance boost using the *HesSTIP* detector, with an increase in average precision of more than 3%. However, it turns out that in combination with *HesSTIP* all descriptors perform almost equally well on the Hollywood dataset. We therefore stick to our default setting of *HesSTIP+stSURF20* for the remainder of this paper.

Drinking and Smoking We report classification accuracy for both actions on the original DrinkingSmoking (DS) dataset as well as the extended (EDS) dataset. The results in figure 3 show that the latter does not improve the classification performance, which stays on par or



(a) ROC



(b) ROC

	DS	EDS
drinking-vs-rest	97.1%	95.5%
smoking-vs-rest	88.7%	89.1%
drinking-vs-smoking	99.1%	99.0%

(c) Average precision

Figure 3: ROC curves for (a) the original DrinkingSmoking (DS) and (b) the Extended-DrinkingSmoking (EDS) dataset. (c) Average precision for both datasets.

even drops by 1.6% in case of the drinking action. This drop can be explained by the nature of the samples that were added. The original dataset contains mainly stationary drinking actions whereas a large number of the added samples show people drinking while walking or were shot with a moving camera. The new annotations contain more diverse motion patterns and thus increase the intraclass variability. The added smoking actions on the other hand do not show this issue and therefore the performance increases slightly. Nonetheless, we score well in all three classification tasks in terms of equal error rate and average precision, albeit not as well as some results reported in [12] (equal error rates for drinking-vs-random between 0.91 and 0.98; no results reported for smoking-vs-random). However, one has to be careful in comparing these values, as it is not clear whether the "random motion patterns" they generated as negative annotations are comparable with our random negative annotations extracted from the training videos. For the task of distinguishing between drinking and smoking actions we clearly outperform [12], who report equal error rates between 0.46 and 0.85.

4.2 Detection results

Next, we report the detection results for 'Drinking' and compare them to state-of-the-art results published in the literature[3]. Two episodes from the movie "Coffee and Cigarettes" are used as test set and contain together 38 drinking actions. Roughly 6.5 million *HesSTIP* features are extracted from the test video of 24 minutes, i.e. an average of 181 features per frame. Using the 50 most discriminative words, they create between $1.0-2.0$ million hypotheses for which it is necessary to compute the confidence (cf. table 2). While this may seem a lot, we merely need to check around 50 hypotheses per frame, instead of an exhaustive sliding window search in 5 dimensions. Detection and description of the features takes approximately 3.5 seconds/frame (with dimension 1024×576) using a parallel implementation with 4 threads on a Quad-Core AMD Opteron™. The action detection runs at about 1 second/frame, where most of the time is spent on the generation of the hypotheses.

In figure 4, we show our detection results together with the results from [12] with and without their keyframe-priming preprocessing step (denoted by OF5Hist-KFtrained and OF5GradHist respectively). With an average precision on the DrinkingSmoking dataset of 45.2%, we perform significantly better than the 12.5% and 42.2% reported in [12]. Furthermore, we obtain higher precision at lower recall which in many applications is desirable.

Although exemplar-based approaches generally perform better the more exemplars are

[3] 'Smoking' is omitted due to lack of comparison with [12].

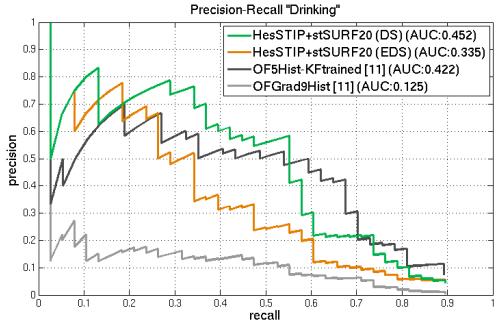


Figure 4: Precision-recall curves for 'Drinking' using the DrinkingSmoking (DS) and ExtendedDrinkingSmoking (EDS) datasets. The average precision is computed as the area under the curve (AUC).

available, our detection performance drops on the extended dataset (EDS). This is consistent with our observations in the context of classification in section 4.1. The extended dataset generates more hypotheses since it contains more exemplars (see table 2), but the corresponding bounding boxes vary more in terms of aspect ratio and scale due to *e.g.* camera motion. Thus the extra hypotheses are not as accurate as the original ones. As a result, they do not improve the detection rate, yet allow for more false positives to sneak in.

For better illustration, we show the first 15 detections for 'Drinking' on the original dataset (DS) in figure 5. Some of the high ranking false positives may not be very intuitive at first (*e.g.* patches of smoke). Those detections are not very similar in appearance to typical drinking actions, yet contain similar bag-of-words. These errors indicate potential further refinements to the pipeline, such as improving the SVM via bootstrapping or adding spatial information.

5 Conclusion and future work

We presented in this paper an approach for action detection in video without having to resort to exhaustive sliding window methods or high dimensional voting spaces. This was

Figure 5: The top 15 detection results for 'Drinking' trained on the original dataset (ordered top to bottom, left to right). True positives are shown in green, false positives in red.

	#hypotheses after 1^{st} pruning stage	after 2^{nd} pruning stage	#detections	#found
DS	1022313 (28 per frame)	196351 (19% of 1^{st} stage)	802	34
EDS	2231215 (61 per frame)	227565 (10% of 1^{st} stage)	732	34

Table 2: Statistics of the detection of the 'Drinking' action for both datasets: the number of hypotheses after the first and second pruning stage, the final number of detections and the number of actions found (out of 38).

achieved by extending the exemplar-based object detection work of Chum *et al*. [3] to the spatio-temporal domain. Starting from local, dense, scale-invariant spatio-temporal features, we select the most discriminative visual words and use these to cast bounding box hypotheses. Detections are obtained by merging those hypotheses with a high confidence value. Although the approach has been stripped of any refinements that may boost performance further, the results clearly demonstrate its viability. One can immediately envision several of these refinements: fine-tuning the parameters used by the system, using spatial pyramids instead of simple bag-of-words, bootstrapping the SVM by using the false positive detections, etc. Yet, even without these enhancements, we already obtain state-of-the-art results.

Acknowledgements

This work is supported by the European IST Programme DIRAC Project FP6-0027787, the IWT SBO-060051 project AMASS++, IBBT and the Flemish Fund for Scientific Research (FWO).

References

[1] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 882–888, 2004.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110:346–359, 2008.

[3] O. Chum and A. Zisserman. An exemplar model for learning object classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[4] M. Dimitrijevic, V. Lepetit, and P. Fua. Human body pose recognition using spatio-temporal templates. In *ICCV workshop on Modeling People and Human Interaction*, 2005.

[5] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *International workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005.

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results. http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.

[7] D.M. Gavrila. A bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8), 2007.

[8] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247–2253, December 2007.

[9] A. Kläser, M. Marszalek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *Proceedings British Machine Vision Conference*, 2008.

[10] C. H. Lampert, M. B. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. In *Proceedings CVPR08*, 2008.

[11] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64 (2-3):107–123, 2005.

[12] I. Laptev and P. Perez. Retrieving actions in movies. In *Proceedings ICCV07*, 2007.

[13] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Proceedings CVPR08*, 2008.

[14] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289, 2008.

[15] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[16] K. Mikolajczyk and H. Uemura. Action recognition with motion-appearance vocabulary forest. In *Proceedings CVPR08*, 2008.

[17] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[18] A. F. Smeaton, P. Over, and W. Kraaij. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In Ajay Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*, pages 151–174. Springer Verlag, Berlin, 2009. ISBN 978-0-387-76567-9.

[19] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Conditional models for contextual human motion recognition. In *ICCV*, pages 1808–1815, 2005.

[20] J. Sullivan and S. Carlsson. Recognizing and tracking human action. In *Proceedings ECCV02*, 2002.

[21] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *Proceedings ECCV08*, 2008.

[22] S.-F. Wong, T.-K. Kim, and R. Cipolla. Learning motion categories using both semantic and structural information. In *CVPR'07*, pages 1–6, 2007.