Using Action Objects Contextual Information for a Multichannel SVM in an Action Recognition Approach based on Bag of Visual Words

Jordi Bautista-Ballester[1,2], Jaume Vergés-Llahí[1] and Domenec Puig[2]
[1]*ATEKNEA Solutions, Víctor Pradera, 45, 08940 Cornellà de Llobregat, Spain*
[2]*Department of Computer Engineering and Mathematics, Universitat Rovira i Virgili, Tarragona 43007, Spain*

Keywords: Action Recognition, Bag of Visual Words, Multikernel Support Vector Machines, Video Representation.

Abstract: Classifying web videos using a Bag of Words (BoW) representation has received increased attention due to its computational simplicity and good performance. The increasing number of categories, including actions with high confusion, and the addition of significant contextual information has lead to most of the authors focusing their efforts on the combination of descriptors. In this field, we propose to use the multikernel Support Vector Machine (SVM) with a contrasted selection of kernels. It is widely accepted that using descriptors that give different kind of information tends to increase the performance. To this end, our approach introduce contextual information, i.e. objects directly related to performed action by pre-selecting a set of points belonging to objects to calculate the codebook. In order to know if a point is part of an object, the objects are previously tracked by matching consecutive frames, and the object bounding box is calculated and labeled. We code the action videos using BoW representation with the object codewords and introduce them to the SVM as an additional kernel. Experiments have been carried out on two action databases, KTH and HMDB, the results provide a significant improvement with respect to other similar approaches.

1 INTRODUCTION

The number of videos uploaded online is increasing every day and recently the analysis of their content has become an intense field of research. In this context, our research focuses on the recognition of action in videos containing contextual information about the means by which an action is carried out. Initially, the sort of databases over which the actions were conformed a set of videos where scenes and parameters such as illumination, focus, distance, and viewpoints were mostly controlled, and few or none data existed on the tools and objects that were involved in the action. For example, the KTH database (Schuldt et al., 2004), a popular choice to test different action recognition techniques, has not such kind of information. In any case, we use this database in the present work to show the performance of our approach in comparison to the rest of other state-of-the-art approaches.

Recently, however, more realistic databases have increasingly been employed in order to go beyond the current state of the art. These sets include videos that stage more realistic actions. A relevant database, HMDB (Kuehne et al., 2011), is one of the largest action video database to-date with 51 action categories, which in total contain almost 7,000 manually annotated clips extracted from a variety of sources ranging from digitized movies to YouTube. This database has been created to evaluate the performance of computer vision systems for action recognition and explore the robustness of these methods under various conditions such as cluttered backgrounds, fast irregular motions, occlusions and camera motion. Several approaches have been proposed in the literature for the recognition of actions in diverse real-world videos. In this database, actions that are contextually connected to the tool or object employed in their performance can be found.

In order to increase the robustness of the recognition of actions in more challenging situations, we propose an approach that is able to integrate two sources of information relevant to discriminate actions, namely, the space-time data that describes the motion and the contextual information that explains how the action is carried out. Specifically, by using the HMDB (Kuehne et al., 2011) database, we select a subset of actions that are performed using a tool or object, a contextual information that allows to discriminate apparently similar actions such as shooting a gun or a bow, which its biggest difference lies in

Bautista-Ballester J., Vergés-Llahí J. and Puig D..
Using Action Objects Contextual Information for a Multichannel SVM in an Action Recognition Approach based on Bag of VisualWords.
DOI: 10.5220/0005301000780086
In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications* (VISAPP-2015), pages 78-86
ISBN: 978-989-758-090-1

the object employed to carry out the action. We explain how these different sources of information can be combined in richer description of human actions that permits higher recognition rates.

The main contributions of this paper, in contrast to other approaches which can be found in recent surveys (Poppe, 2010; Weinland et al., 2011), are the introduction of contextual information of actions into BoW-based descriptors and a recognition structure that allows the addition of new information using multichannel SVM (Zhang et al., 2006). Multichannel SVM has previously proven very successful in action recognition (Wang et al., 2013; Bilinski and Corvee, 2013) and we take advantage of this structure by adding data which is strictly not a descriptor of motion but contextual information describing the tool employed in the action, which is a new way of using multichannel SVM.

With respect to the type of action descriptor, local space-time features (Dollar et al., 2005; Laptev, 2005) have shown to be successful for general action recognition because they avoid non-trivial pre-processing steps, such as tracking and segmentation, and provide descriptors invariant to illumination and camera motion. In particular, HOG3D (Kläser et al., 2008) has proven to outperform most of this sort of descriptors (Willems et al., 2008; Scovanner et al., 2007). Another approach has been trying to find the best combinations of different simpler descriptors. To this end, (Snoek et al., 2005) studied the different methods of descriptor fusion and classified them into *early* or *late* fusion approaches. The former one consists in a fusion before the training step, while the latter is a fusion afterwards.

Concerning the combination of features, (Ikizler-Cinbis and Sclaroff, 2010) combines six different descriptors for three different contextual information, namely, *people* (HOF and HOG3D), *objects* (HOF and HOG), and *scene* (GIST and color histograms). Their combination is accomplished using a multiple MIL approach, which is a concatenation of bag representations and classified with an L2-Regularized Linear SVM. On the other hand, (Bilinski and Corvee, 2013) uses relative dense *tracklets* for action recognition. They compute two specific descriptors, SMST and RMST, in order to obtain information from the actions relative to the head of the performer. Two more descriptors encoding space and time, HOG and HOF, are employed. A multichannel χ^2 kernel SVM is used for the combination of this set of descriptors. Similarly, (Wang et al., 2011; Wang and Schmid, 2013) compute dense trajectories and their descriptors –HOG, HOF, and MBH– to finally combine them using a multichannel SVM. In contrast, using a late fusion of the descriptors the approach in (Reddy and Shah, 2013) trains a SVM for a scene context descriptor and another SVM for a motion descriptor, using a histogram intersection kernel. The two probability estimates obtained separately from each SVM are fused into one single recognition output afterwards.

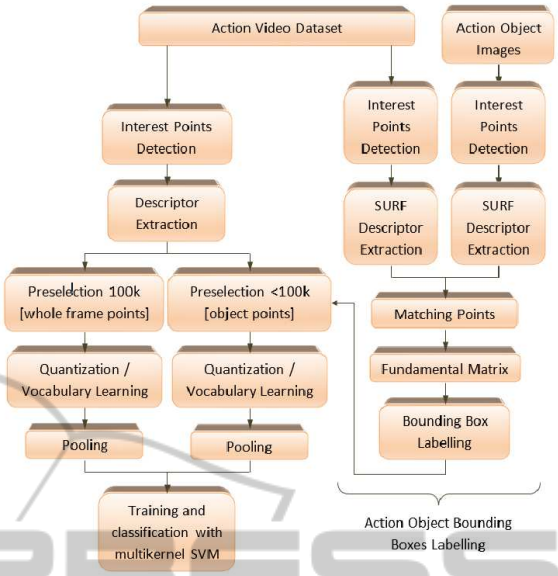


Figure 1: Scheme overview in the proposed approach.

In our work, we use information describing the object involved in an action using a BoW based action recognition approach. To this end, we first detect the set of points belonging to the object by matching them to an instance of the object. This process also labels the bounding boxes, which are later used to compute a new *codebook* –the dictionary employed to compute the relative frequencies in a BoW description–, and the information about the objects in the actions is preserved as a consequence. Afterwards, we employ such codebook to encode the video frames computing a BoW description. Finally, we combine the two source of information, motion and context, by means of a multikernel SVM. Experimental results show that this procedure improves the recognition of actions.

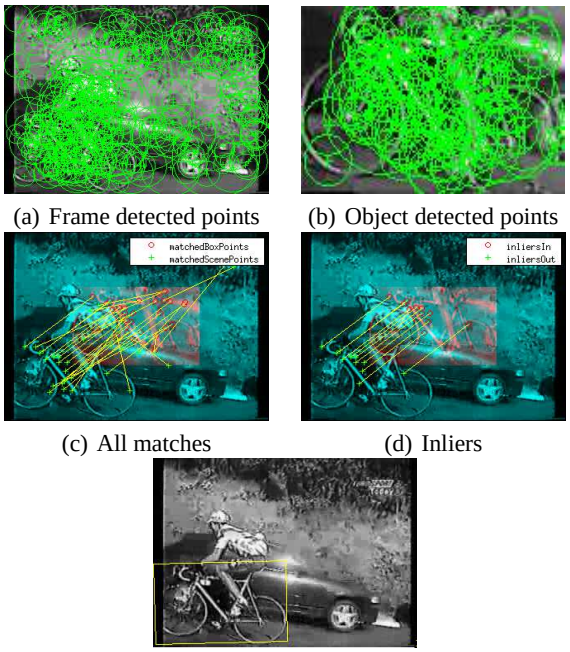
The rest of the paper is organized as follows. In Section 2, we detail our approach for action object detection and the method employed to label the bounding box around contextual information. In this section we also show how to use these labels during posterior codebook generation. Experimental setup and the two databases used to evaluate our method is explained in Section 3. In Section 4, we present our experimental results over the two databases. Finally, in Section 5 we discuss the results and conclude with future directions of the work.

2 SCHEME OVERVIEW

The main goal of our method is to introduce object information relevant to the action into the BoW based representation of action. In Section 2.1 we explain the method employed to detect and track the objects in the video frames and extract the bounding box enclosing the object as well as the way to label the object. In Section 2.2 we consider the procedure to add this object information into the training system. A flowchart describing our approach is shown in Fig. 1.

2.1 Object Detection and Tracking

In order to add contextual information related to the object appearing in an action, we must find the object in the video sequence. Each video contains one action, and we detect one object per action. Therefore, we obtain one instance image of each object per video and use this image to find the object along the whole video by matching a set of points previously extracted from the frame and the instance image. The matching procedure, based on the epipolar geometry described in (Hartley and Zisserman, 2004), is described in Fig. 2. The points are extracted using a Harris corner detector and by SURF features. This way ensures a large set of points belonging to the object,



(a) Frame detected points (b) Object detected points

(c) All matches (d) Inliers

(e) Bounding box

Figure 2: First row: point detection and descriptor extraction for a video frame and the object image. Second row: matches and outlier filtering. Third row: transformed bounding box.

which is necessary to obtain good point correspondences and compute a representative bounding box. Then, we compute the point matching applying the Nearest Neighbor algorithm and set a threshold to select the strongest matches.

Finally, we compute the fundamental matrix – excluding outliers by using Random Sample Consensus (RANSAC)– and use it to obtain a transformation of the initial bounding box. This ensures more accuracy around the area that limits the object in the frame. The result of this procedure is a bounding box enclosing the object used in each action for each frame in the video as can be seen in Fig. 2.

In order to add this information to the overall scheme, we first extract Space-Time Interest Points (STIP) (Laptev, 2005) from each frame and video and compute their descriptor. Next, we select a maximum of 100k of object points applying the bounding boxes labels. Then, we construct a codebook from the preselected words belonging to objects and combine this codebook with others using the multikernel SVM explained in the following section.

2.2 Multikernel for SVM

Visual features extracted from a video can represent a wide variety of information, such as scene (e.g., GIST (Solmaz et al., 2012)), motion (e.g., HOF (Lucas and Kanade, 1981), MBH (Dalal et al., 2006), HOG3D (Kläser et al., 2008)) or even just color (color histograms). To classify actions using all these features the information must be fused in an appropriate way. According to the moment of the combination, (Snoek et al., 2005) proposed a classification of the fusion schemes in *early* or *late* fusion. In early fusion the descriptors are combined before training a classifier (e.g., concatenating (Ikizler-Cinbis and Sclaroff, 2010; Reddy and Shah, 2013)), and in late fusion the classifiers are trained for each descriptor and the fusion is done for the results of all these classifiers (e.g., probabilistic fusion (Reddy and Shah, 2013)).

We use an early fusion in our approach since the combination is done before the training. A SVM with a chi-squared kernel is used for classification,

$$\chi^2(h_i,h_j) = \frac{1}{2}\sum_{k=1}^{n}\left(\frac{(h_i(k)-h_j(k))^2}{h_i(k)+h_j(k)}\right) \qquad (1)$$

fusing all different descriptors by summing their kernel matrices normalized by the average distance.

$$K(h_i,h_j) = exp(-\sum_c \frac{1}{A^c}\chi^2(h_i^c,h_j^c)) \qquad (2)$$

The value of A^c is the mean value of χ^2 distances between the training samples for the c-th channel

Table 1: Descriptors used to encode frames.

Descriptor	Characteristics	Reference
trajectories	KLT tracker or SIFT matcher	(Jiang et al., 2012)
HOG	static appearance information from local gradients	(Dalal and Triggs, 2005)
HOF	local motion information	(Lucas and Kanade, 1981)
MBH	separately computes vertical and horizontal OF components	(Dalal et al., 2006)
HOG3D	spatio-temporal extension of HOG	(Kläser et al., 2008)

(Zhang et al., 2006). All the weights are set to one, meaning that none of the kernels is more discriminative than the others.

3 EXPERIMENTAL SETUP

In this section object detection and tracking are considered in detail. Afterwards, we introduce the feature encoding in our evaluation step. Finally, the databases and their experimental setup are exposed.

3.1 Object Detection

The points used to identify and track the objects are a mixture of points obtained with Harris corner detector and features computed applying SURF. We use a threshold between 0,04 and 0,1 for Harris detector and a maximum number of 1000 points for SURF. This ensures enough quantity of points with enough quality belonging to the object, even in the case the object appearing in the video sequence is relatively small, like a ball or a sword. For the matching, we select the strongest 1% matches, which is restrictive but ensures better point correspondences.

3.2 BoW based Encoding

To encode frames, we use the BoW approach. First, we make use of STIP points following the work in (Laptev, 2005). We compute different descriptors for each point, namely, HOG3D, trajectories, HOG, HOF, and MBH. In the case of HOG3D descriptors, we set the parameters optimized for the KTH database as described in (Kläser et al., 2008), resulting in 1008 dimensions in total. In the case of trajectories, HOG, HOF, and MBH, we follow the work of (Wang et al., 2011) and set the parameters as they did. The dimensions of these descriptors are, respectively, 30 for trajectories, 96 for HOG, 108 for HOF and 192 for MBH, which are significantly smaller than HOG3D. DENSE_T is obtained as the concatenation of trajectories, HOG, HOF and MBH, which represents an *early* fusion and its dimension is 426. We train a codebook for each descriptor type using a maximum of 100k randomly sampled features. For the object kernel, we ensure the object point selection using the method described in Section 2.

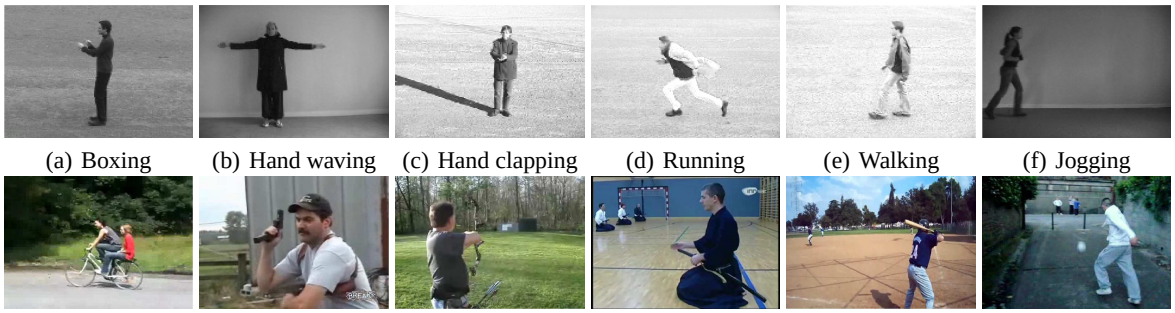
Afterwards, we group the points employing the k-Means clustering algorithm with a maximum of 5 iterations. The size of the codebook is set to 500 words, following (Reddy and Shah, 2013)(Bilinski and Corvee, 2013) where the codebook size is limited to 500 or 1000 to avoid over-learning and despite the fact that the larger the number of clusters employed, the better the performance is. Finally, a SVM with an exponential chi-squared kernel is used for classification, combining all different descriptors by summing their kernel matrices and normalizing the result by the average distance. We use a 10 fold cross-validation with one-against-all approach. For all the experiments we employ the default parameter values in the *libsvm* library (Chang and Lin, 2011).

3.3 Databases Used in the Experiments

As previously said, we test our approach with two different databases, KTH and HMDB. KTH is database by (Schuldt et al., 2004) that does not contain any tool or object related to any action. Despite we can not take advantage of any contextual data, this experimentation allows us to test whether our approach is comparable to these of the state of the art. HMDB database is collected by (Kuehne et al., 2011) and is a more challenging and realistic one where objects used in actions are present.

3.3.1 KTH Database

The KTH database (Schuldt et al., 2004) consists of 6 actions performed by 25 actors in a structured homogeneous environment with a total of 598 videos. The actions performed are *boxing*, *hand-waving*, *hand-clapping*, *running*, *walking* and *jogging*, with no object involved in any of these actions. In order to reduce the computational burden, we pre-select 12 videos for any action performed by randomly se-



(a) Boxing (b) Hand waving (c) Hand clapping (d) Running (e) Walking (f) Jogging

(g) Ride bike (h) Shoot gun (i) Shoot bow (j) Draw sword (k) Swing baseball (l) Kick ball

Figure 3: Example frames from KTH database (first row) and HMDB database (second row). We use all the actions in KTH, that is, (a) boxing, (b) hand waving, (c) hand clapping, (d) running, (e) walking, (f) jogging, and a subset of the 51 actions in HMDB that include objects, (g) ride bike, (h) shoot gun, (i) shoot bow, (j) draw sword, (k) swing baseball, and (l) kick ball.

lected actors into different environments, ensuring that as many variation as possible are employed, i.e., scene, person, illumination and camera distance, which makes a total of 72 videos.

3.3.2 HMDB Database

The HMDB database (Kuehne et al., 2011) consists of 51 actions from a total of 6,849 videos collected from a variety of sources ranging from digitized movies to YouTube videos. The action categories are grouped in five types: general facial actions, facial actions with object manipulation, general body movements, body movements with object interaction, and body movements for human interaction.

Considering that we need actions with object interaction, we do not follow the original splits proposed by (Kuehne et al., 2011). We reduce the computational cost by pre-selecting 6 different actions with 20 videos per action, resulting in 120 videos in total. The pre-selected actions are *ride bike*, *shoot gun*, *shoot bow*, *draw sword*, *swing baseball* and *kick ball*. The purpose of this selection is dual: first, ensuring that an object is involved in the action, and second, ensuring the presence of as many variations as possible. Similar actions are also taken into account, a fact that makes the set more challenging.

In order to ensure the presence of as many variations as possible, we follow a proportion of clips similar to that in the complete database. The whole set of videos corresponding to these 6 actions has a 63.44% of actions showing the *full body*, a 32.51% showing the *upper body*, a 2.46% the *head*, and a 1.59% the *lower body*. The set we selected has a proportion of 63.33%, 32.5%, 2.5% and 1.67% respectively. We also maintain the same proportions for the number of people involved (*1*,*2*, *other*), camera motion (*motion*, *no motion*), camera viewpoint relative to the author (*front*, *back*, *left*, *right*) and for the video quality (*bad*, *medium*, *good*). All the values of these proportions can be seen in Table 2.

4 EXPERIMENTAL RESULTS

We first analyze the use of multikernel SVM in Section 4.1. We want to know whether there is a difference in using a single kernel or a multiple kernel. Also, we compare the effect of different combinations of descriptors. In Section 4.2 we evaluate the impact of the addition of contextual information, based on the detection of the object related to the action.

Table 2: HMDB subset selection. We maintain proportions with respect to the original set of videos for the same actions: ride bike, shoot gun, shoot bow, draw sword, swing baseball, and kick ball.

		Original set (%)	our own set (%)
part of body	f	63.44	63.33
	h	1.59	2.5
	l	2.46	1.67
	u	32.51	32.5
♯ people	np1	92.77	90.83
	np2	4.19	3.33
	npn	3.04	5.84
camera motion	cm	52.46	60.83
	nm	47.54	39.17
camera viewpoint	ba	18.06	20
	fr	49.28	46.66
	le	16.91	16.67
	ri	15.75	16.67
video quality	bad	19.80	21.67
	goo	8.24	9.16
	med	71.96	69.17
♯ videos		692	120

Table 3: Comparison of different descriptors on the databases using our approach.

Databases	KTH (%)	HMDB (%)	HMDB + obj (%)	Δ (%)
trajectories	45.51	38.13	39.81	1.68
HOG	70.63	54.29	64.76	10.47
HOF	62.99	41.67	44.78	3.11
MBH	61.55	38.3	47.10	8.8
DENSE_T	72.42	45.81	53.99	8.18
HOG3D	**86.57**	71.98	**79.58**	7.6

4.1 Channel Selection

The use of a multikernel SVM allows us to add different descriptors into the traditional BoW approach for action recognition. This approach permits to include several descriptors into this scheme as explained in (Wang et al., 2011), where a combination of trajectories, HOG, HOF, and MBH is employed, and analyze how their combination by means of a multikernel SVM improves the performance with respect to any singular descriptor. In our work we do the same for a different set of descriptors, including trajectories, HOG, HOG, MBH, DENSET (an early fusion of them) and HOG3D. Results for all these descriptors using our approach can be seen in Table 3.

In our procedure, we have chosen a first descriptor and have progressively added new ones in order to see the effect of including new information into the kernel. To see the best improvements, we have chosen the descriptor that contributes the least, i.e., trajectories. These results can be seen in Table 4. Initially, this single descriptor gives a performance of 38.13%. Adding a descriptor that contributes more, HOG, the new value is 57.83%, which shows an improvement surpassing a 50% increase. On the other hand, adding another weak descriptor, HOF, the new value becomes 43.0%, which represents an improvement of a 12.8%. This fact shows the importance of choosing a good combination of descriptors. Almost all the additions improve the results, but the question is which one provides the best results since adding new channels results in higher computational costs. Therefore, we want the least number of channels that obtains the best results.

4.2 Evaluation of Adding Contextual Information

In the case of the KTH database, where no objects are available, the present method equals the results of the multichannel approach in (Wang et al., 2011). However, there is a significant improvement in the case where contextual information is present. In that case, our method outperforms the results obtained for all the descriptors, ranging from a minimum increase of 1.68% (HOF) to a maximum of 10.47% (HOG). The same happens when combination of descriptors are used and adding objects to the HOG + trajectories combination generates the highest increase, 13.74%, which also outperforms the rest of combinations. The highest value for each database is highlighted in boldface in Table 4.

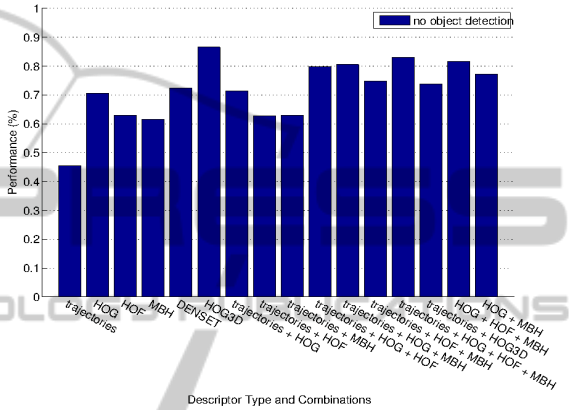


Figure 4: Evaluation of our approach for the KTH database.

The idea of contextual information influence can be seen in the confusion matrices in Fig. 6. For example, *shoot bow* has confusions with the rest of actions, that is, 7% with *ride bike*, 17% with *draw sword*, 29% with *shoot gun*, 22% with *swing baseball*, and 5% with *kick ball*. After adding object information, these values are all reduced: 5% with *ride bike*, 15% with *draw sword*, 24% with *shoot gun*, 20% with *swing baseball* and 5% with *kick ball*, which means that the confusion of this action with respect to the rest

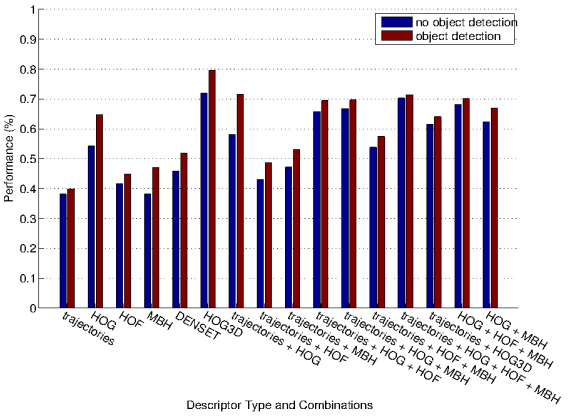


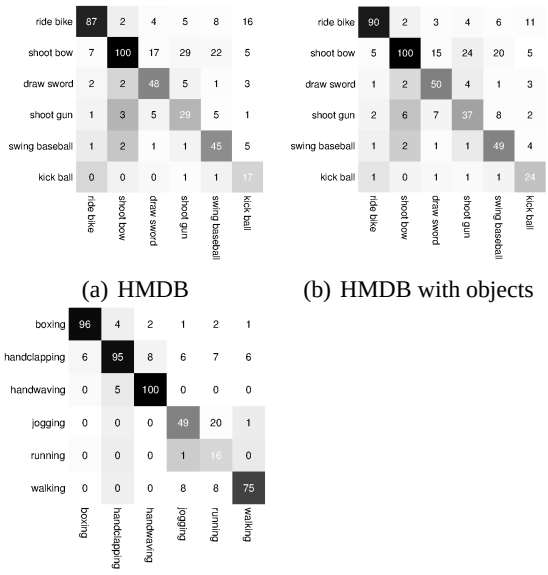
Figure 5: Evaluation of our approach using object detection for the HMDB database.

Table 4: Comparison of different descriptors combinations on the databases with our approach.

Databases	KTH (%)	HMDB (%)	HMDB + obj (%)	Δ (%)
trajectories + HOG	71.33	57.83	**71.57**	13.74
trajectories + HOF	62.77	43.00	48.64	5.64
trajectories + MBH	62.98	45.38	52.99	7.61
trajectories + HOG + HOF	79.83	64.67	69.39	4.72
trajectories + HOG + MBH	80.5	66.45	69.66	3.21
trajectories + HOF + MBH	74.68	53.44	57.45	4.01
trajectories + HOG + HOF + MBH	**82.94**	70.04	72.97	2.93
trajectories + HOG3D	73.67	61.56	64.07	2.51
HOG + HOF + MBH	81.66	68.09	70.23	2.14
HOG + MBH	77.13	60.39	66.92	6.53

is smaller as a consequence of including contextual information into the action description.

HOG3D and DENSET descriptors are used here to show two significant facts. First, that using a unique optimal descriptor is better than a combination of several descriptors that individually perform worse. This is apparent by the fact that HOG3D, which fuses information of space and time in a single descriptor, obtains a 71.98%. This result cannot be reached either by a concatenation of descriptors –trajectories, HOG, HOF, and MBH– or by a multikernel combination of the same descriptors, despite the latter is almost as good as HOG3D and reaches a performance of 70.04% while the former can at most get a value of 45.81%. Moreover, despite that no combination can outperform the best results reached by HOG3D, the addition of object information is able to increase the HOG3D result an extra 7.6%, up to 79.58%. Therefore, it is clearly stated that including contextual information always results in an improvement.



(a) HMDB (b) HMDB with objects

(c) KTH

Figure 6: Confusion matrix for the (a) HMDB database using trajectories, HOG, HOF, MBH descriptors as it is done in (Wang et al., 2011) with average performance for 500 codewords: 68.09%, (b) HMDB with our approach using the same configuration as (a), with average performance for 500 codewords: 70.23%, and (c) confusion matrix for the KTH database using trajectories, HOG, HOF, MBH descriptors as it is done in (Wang et al., 2011). Average performance for 1000 codewords: 81.66%

Secondly, that combining descriptors is something that should be done with adequate criteria: Tables 3 and 4 show that early combination as a concatenation perform worse (45.81%) than using a late composition of trajectories, HOG, HOF, and MBH (70.04%) using a multikernel SVM. Figures 4 and 5 summarize all these results.

From the results obtained in this Section, we can state there is a clear improvement in the action recognition task as a consequence of including contextual information in the action description and recognition. Moreover, the present paper shows a method that allows the obtaining and addition of such information.

5 CONCLUSIONS AND FUTURE WORK

In this paper we have proposed a method to incorporate action contextual information that extends a previous method used to combine motion related information into a standard action recognition scheme based on BoW. This approach allows the addition of information related to the tool or object employ in the execution of an action and shows an increment of the overall recognition performance. We have shown that adding information without any specific purpose might lead to a lack of improvement adding the consequent computational cost to the scheme. Our approach complements space and time information and

proposes a procedure to add any sort of contextual information that can be further generalized to include other data apart from the object used during an action. Additionally, the present approach shows that the best results are obtained when kernels from spacial, temporal, and tool information are combined into a multichannel SVM kernel. In this respect, the highest recognition rates are 71.57% using a combination of trajectories, HOG and object. In the near future we plan to add more contextual information –scene– in order to improve the results.

ACKNOWLEDGEMENTS

This research has been partially supported by the Industrial Doctorate program of the Government of Catalonia, and by the European Community through the FP7 framework program by funding the Vinbot project (N 605630) conducted by Ateknea Solutions Catalonia.

REFERENCES

Bilinski, P. and Corvee, E. (2013). Relative Dense Tracklets for Human Action Recognition. *10th IEEE International Conference on Automatic Face and Gesture Recognition*.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA. IEEE Computer Society.

Dalal, N., Triggs, B., and Schmid, C. (2006). Human detection using oriented histograms of flow and appearance. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part II*, ECCV'06, pages 428–441, Berlin, Heidelberg. Springer-Verlag.

Dollar, P., Rabaud, V., Cottrell, G., and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. In *Proceedings of the 14th International Conference on Computer Communications and Networks*, ICCCN '05, pages 65–72, Washington, DC, USA. IEEE Computer Society.

Hartley, R. I. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition.

Ikizler-Cinbis, N. and Sclaroff, S. (2010). Object, scene and actions: Combining multiple features for human action recognition. In *Proceedings of the 11th European Conference on Computer Vision: Part I*, ECCV'10, pages 494–507, Berlin, Heidelberg. Springer-Verlag.

Jiang, Y., Dai, Q., Xue, X., Liu, W., and Ngo, C. (2012). Trajectory-based modeling of human actions with motion reference points. In *European Conference on Computer Vision (ECCV)*.

Kläser, A., Marszałek, M., and Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference*, pages 995–1004.

Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T. (2011). HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Laptev, I. (2005). On space-time interest points. *Int. J. Comput. Vision*, 64(2-3):107–123.

Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, pages 674–679, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Poppe, R. (2010). A survey on vision-based human action recognition. *Image Vision Comput.*, 28(6):976–990.

Reddy, K. K. and Shah, M. (2013). Recognizing 50 human action categories of web videos. *Mach. Vision Appl.*, 24(5):971–981.

Schuldt, C., Laptev, I., and Caputo, B. (2004). Recognizing human actions: A local svm approach. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 3 - Volume 03*, ICPR '04, pages 32–36, Washington, DC, USA. IEEE Computer Society.

Scovanner, P., Ali, S., and Shah, M. (2007). A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th International Conference on Multimedia*, MULTIMEDIA '07, pages 357–360, New York, NY, USA. ACM.

Snoek, C. G. M., Worring, M., and Smeulders, A. W. M. (2005). Early versus late fusion in semantic video analysis. In *Proceedings of the 13th Annual ACM International Conference on Multimedia*, MULTIMEDIA '05, pages 399–402, New York, NY, USA. ACM.

Solmaz, B., Modiri, S. A., and Shah, M. (2012). Classifying web videos using a global video descriptor. *Machine Vision and Applications*.

Wang, H., Kläser, A., Schmid, C., and Liu, C. (2013). Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*.

Wang, H., Kläser, A., Schmid, C., and Liu, C.-L. (2011). Action Recognition by Dense Trajectories. In *IEEE Conf. on Computer Vision & Pattern Recognition*, pages 3169–3176, Colorado Springs, United States.

Wang, H. and Schmid, C. (2013). Action Recognition with Improved Trajectories. In *ICCV 2013 - IEEE International Conference on Computer Vision*, pages 3551–3558, Sydney, Australie. IEEE.

Weinland, D., Ronfard, R., and Boyer, E. (2011). A survey of vision-based methods for action representation,

segmentation and recognition. *Computer Vision Image Understanding*, 115(2):224–241.

Willems, G., Tuytelaars, T., and Gool, L. (2008). An efficient dense and scale-invariant spatio-temporal interest point detector. In *Proceedings of the 10th European Conf. on Computer Vision: Part II*, ECCV '08, pages 650–663, Berlin, Heidelberg. Springer-Verlag.

Zhang, J., Marszalek, M., Lazebnik, S., and Schmid, C. (2006). Local features and kernels for classification of texture and object categories: A comprehensive study. In *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, CVPRW '06, pages 13–, Washington, DC, USA. IEEE Computer Society.