Temporal Segmentation of Human Actions in Video Sequences

Josep Maria Carmona, Joan Climent
Barcelona Tech (UPC), Barcelona, Spain

***Abstract*— Most of the published works concerning action recognition, usually assume that the action sequences have been previously segmented in time, that is, the action to be recognized starts with the first sequence frame and ends with the last one. However, temporal segmentation of actions in sequences is not an easy task, and is always prone to errors. In this paper we present a new technique to automatically extract human actions from a video sequence.**

Our approach presents several contributions. First of all, we use a projection template scheme and find spatio-temporal features and descriptors within the projected surface, rather than extracting them in the whole sequence. For projecting the sequence we use a variant of the R transform, which has never been used before for temporal action segmentation. Instead of projecting the original video sequence, we project its optical flow components, preserving important information about action motion.

We test our method on a publicly available action dataset, and the results show that it performs very well segmenting human actions compared with the state-of-the-art methods.

Keywords—temporal segmentation; PHOW; R transform

I. INTRODUCTION

Automatic recognition of human actions has attracted the interest of the pattern recognition and computer vision research communities in recent years. Applications of visual action recognition include gestural human-computer interaction, video surveillance, biometrics analysis, or disabled people assistance.

Most of current literature on action recognition is mainly based on the recognition of actions previously segmented in time. In real world applications, actions occur continuously in time, without any identification of beginning or end. Our objective is to find a method to detect and determine the moment where a human action is taking place within a video sequence.

Some authors have tried to solve the problem of temporal segmentation. In [1], they propose a unified framework that encodes spatio-temporal relationships among moving parts and the individual poses appearances. They use an unsupervised learning algorithm that automatically learns pose models and motion parts. The segmentation of periodical human movements in temporal cycles is considered in [9]. They use the Pyramid Correlogram of Oriented Gradients (PCOG) descriptor on Motion History Images (MHI) and Motion Energy Images (MEI) projection templates [15] to obtain shape and movement information. These descriptors are classified using a multi-class SVM with a RBF kernel. Unfortunately, they do not present any results on a public sequence dataset. In [6], authors propose a one-shot (one training sequence for each action) method based on 3D Histograms of Scene Flow (3DHOFs) and Global Histograms of Oriented Gradient (GHOGs) descriptors. They work on RGBD images, and also use SVMs as classifier. A unified method for gesture segmentation and gesture recognition is presented in [7]. They extend the Bag of Words (BoW) paradigm to the temporal domain, and use dynamic programming to maximize the selected class scores of a multi-class SVM output.

In this paper we present a template-based approach for temporal action segmentation. Templates are obtained by a variant of the $\mathbb{R}$ transform. Although the $\mathbb{R}$ transform was originally designed for object recognition, some authors have also used it for action recognition [2,3,4]. Most of these works apply the transform to silhouette images or to the human shapes previously segmented from video frames. This segmentation stage involves all the problematic issues concerning illumination changes, shades, noise... In [23], authors capture the geometrical distribution of interest points extending the $\mathbb{R}$ transform to 3D. Our method is able to segment human actions from a video sequence with no need of a previous shape or silhouette extraction. The $\mathbb{R}$ transform has never been used before for action segmentation.

Using a variant of the $\mathbb{R}$ transform, namely $\mathbb{R}_f$, we compute a projection surface from each video sequence. The projection is computed using the optical flow components of the sequence as input. Next, we find the spatio-temporal descriptors in the projected surface, rather than in the original sequence. For keypoint extraction and feature description we have used the Pyramid Histogram of Visual Words (PHOW) [8]. The PHOW descriptors are computed within a temporal window sliding along the $\mathbb{R}_f$ surface, in order to detect the beginning and end of an action in a sequence. As in most of current literature concerning action recognition, we have also used a SVM as classifier. A detailed description of the whole process is given in section 3.

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes,creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. **DOI: 10.1109/IntelliSys.2017.8324220**

II. THE ℝ TRANSFORM

The Radon transform [9] consists of a multiple angle projection of a given image *I(x,y)*. The result of this projection is an integral line, that is, the cumulative sum of pixel values in all directions. It uses the polar expression of a straight line

$$\rho = x\cos\theta + y\sin\theta. \quad (1)$$

The Radon transform can be expressed mathematically using (2).

$$g(\rho,\theta) = \sum_x \sum_y I(x,y)\delta(x\cos\theta + y\sin\theta - \rho), \quad (2)$$

where *I(x,y)* is the input image, δ is the Dirac function, ρ is the distance from the line to the origin, and θ is the projection direction. The main drawback of the Radon transform is that it is not invariant to translation, scale, or rotation. There exist several approaches to achieve such invariances [10]. In [11], they presented a variant of the Radon transform, the ℝ transform, which is invariant to translation and scale.

The ℝ transform is computed summing all squared values of the Radon transform for all image lines of a given direction θ. It can be expressed using (3).

$$R(\theta) = \sum_\rho g^2(\rho,\theta). \quad (3)$$

The result of the ℝ transform is a function giving the normalized sum of pixel values for all orientations. It maps a 2D image to a 1D signal.

The $\mathbb{R}_f$ transform is a variant of the ℝ transform, with f being a generic function. It can be expressed in its general form:

$$R_f(\theta) = f(g(\rho,\theta)), \quad (4)$$

where $g(\rho,\theta)$ is the Radon transform and f is a function that can be tuned as a parameter, and is useful to adapt the transform to the problem being solved.

For example, $\mathbb{R}_{max}$ substitutes the squared values of the ℝ transform by the supremum of the pixel absolute values. This transform is invariant to translation and, if correctly normalized dividing by the maximum of all pixel values, it is also invariant to scale.

$$R_{\max}(\theta) = \begin{cases} \max_\rho(g(\rho,\theta)) & if \quad R_1 \geq R_2 \\ \min_\rho(g(\rho,\theta)) & if \quad R_1 < R_2 \end{cases}, \quad (5)$$

where $R_1 = \left|\max_\rho(g(\rho,\theta))\right|$ and $R_2 = \left|\min_\rho(g(\rho,\theta))\right|$.

$\mathbb{R}_{dev}$ uses the standard deviation instead of the sum of squared values. It is also invariant to translation.

$$R_{dev}(\theta) = \underset{\rho}{dev}(g(\rho,\theta)). \quad (6)$$

$\mathbb{R}_{mean}$ uses the mean pixel value for each direction. Even though it is pretty similar to the original ℝ transform, it has the advantage of considering the negative values of $g(\rho,\theta)$.

$$R_{mean}(\theta) = \underset{\rho}{mean}(g(\rho,\theta)). \quad (7)$$

The properties of all these $\mathbb{R}_f$ transform are totally dependent on the function f chosen. Apart from their properties concerning invariances, they present different behaviours when applied to images that may contain negative values (like the optical flow images used in this work).

Since an ℝ transform projects a 2D image to a 1D signal, the result of applying the transform to a video sequence is a 2D surface template.

We have tested the four transforms ($\mathbb{R}_{max}$, $\mathbb{R}_{mean}$, $\mathbb{R}_{dev}$ and ℝ) on the sequences taken from the Weizmann dataset. In section 4 we show that the best results are obtained using $\mathbb{R}_{max}$.

III. TEMPORAL ACTION SEGMENTATION

First of all, we project the whole input sequence into a single template using the $\mathbb{R}_f$ transform. Instead of using the raw video sequence as input for the $\mathbb{R}_f$ transform, we apply the $\mathbb{R}_f$ transform to both F_x, F_y components of the optical flow, obtaining two surfaces $\mathbb{R}_{fx}$ and $\mathbb{R}_{fy}$. These surfaces can be considered as spatio-temporal templates defining an action sequence. An example of this surface is shown in Fig.1.a. Since $\mathbb{R}_{fx}$ and $\mathbb{R}_{fy}$ have been computed applying an $\mathbb{R}_f$ transform to each frame of the video sequence, each coordinate in the x axis corresponds to a frame of the video

sequence, while each coordinate in the *y* axis corresponds to the direction θ value of the $\Re_f$ projection (from 0 to 180º).

The optical flow has been computed using the real-time algorithm presented in [14]. We have concatenated $\Re_{fx}$ and $\Re_{fy}$ to obtain a single surface for the classification and training stages. Therefore, we obtain an *F*x360 template for a full sequence, with *F* being the number of frames of the sequence.

Our temporal segmentation technique is based on a sliding window model. We slide an *N*x*M* window over the $\Re_f$ surfaces and perform a recognition process within the window area, where *N* is the window width (i.e. the number of frames in the sequence), and *M* is the θ range that the window spans (i.e. 360º, the whole range of directions). Obviously, the window width *N* has to be narrower than the shortest action to be recognized i.e. $N < F$. We have chosen $N = 25$ for training and recognition stages.

We compute a set of *n* PHOW descriptors $D[d_1,.....,d_n]$ within each window on the $\Re_f$. In the recognition stage, we slide a window with the same size (25x360) over the input surface for each input frame. Fig.1 shows an example of the sliding window over an R_f surface, and PHOW computed within this surface fragment. We have worked on a single scale. The number of keypoints extracted within a window depends on the window size (*NxM)*, the distance (*D)* chosen in the grid of dense SIFT, and the scale size (*Sc)* of the descriptors chosen. We have used *N*=25, *M*=360, *D*=1 and *Sc*=3.

After this computation, we have used Bag of features (BoW) technique. To do this, similar descriptors are clustered using a *k*-means algorithm. The centers of these clusters define a Visual Codebook. For the classification stage, we have trained a SVM for each different action class *q* that we aim to segment. Similar to [6], we have trained a linear SVM for each class using a one-versus-all method. That is, for the class *q*, we consider positive examples all subsequences corresponding to actions A_q of class *q*, and negative examples all the remaining subsequences corresponding to the rest of the classes. At the end, we obtain a set of *Q* SVM linear classifiers, where *Q* is the number of action classes.

In the recognition stage, since the response of class-*q* SVM will be high when it is fed with an input similar to training actions, and the rest of SVMs will give low scores, their output scores are used to determine the limits of each action. Therefore, we expect the highest SVM_q score when the sliding window is over an action A_q on the projected surface, and low scores for the rest of *SVM*. When the window is over a non-action region, or in a region overlapping two different actions, we obtain similar low scores in all SVM outputs because no model is predominant over the rest.

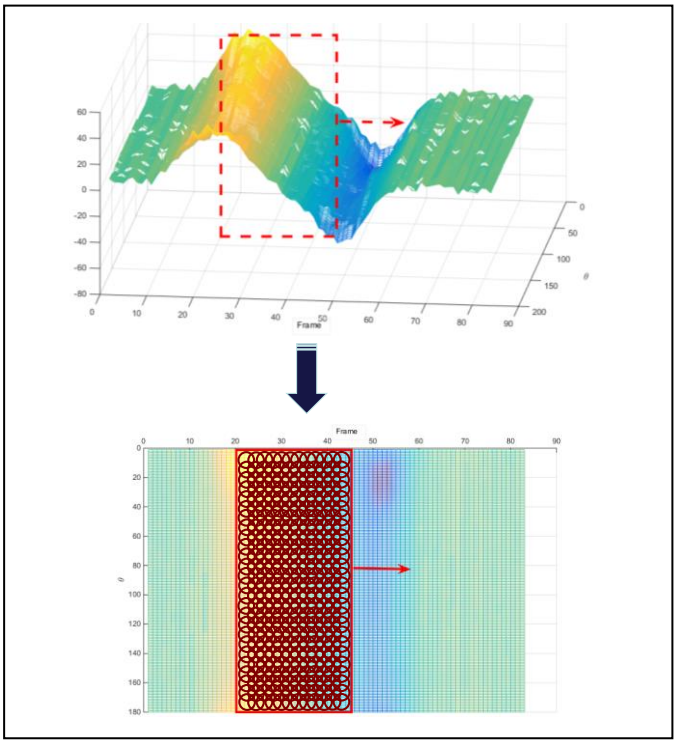

Fig. 1. Example of *a projection* surface computed using $\Re_{max}$ transform on a 'bend' action from the Weizmann dataset. This sequence contains 84 frames. up: sliding window over an $\Re_{max}$ surface. down: PHOW applied over the same $\Re_{max}$ surface. Circles show dense keypoints applied within the window. Not all distances in the grid are shown for visualization purposes.

The scores given by the SVM classifiers are very noisy. We have applied Singular Value Decomposition (SVD) to filter them. We have computed SVD on the SVM outputs, removed all singular values but the first to obtain the filtered signal.

As shown in Fig. 4, the filtered signals obtained can be easily segmented. Salient domes indicate the presence of a given action in the sequence. A local maximum evidences the presence of an action, and dome slopes determine the beginning and end of a given action. Fig. 2 depicts the block diagram of the whole process.

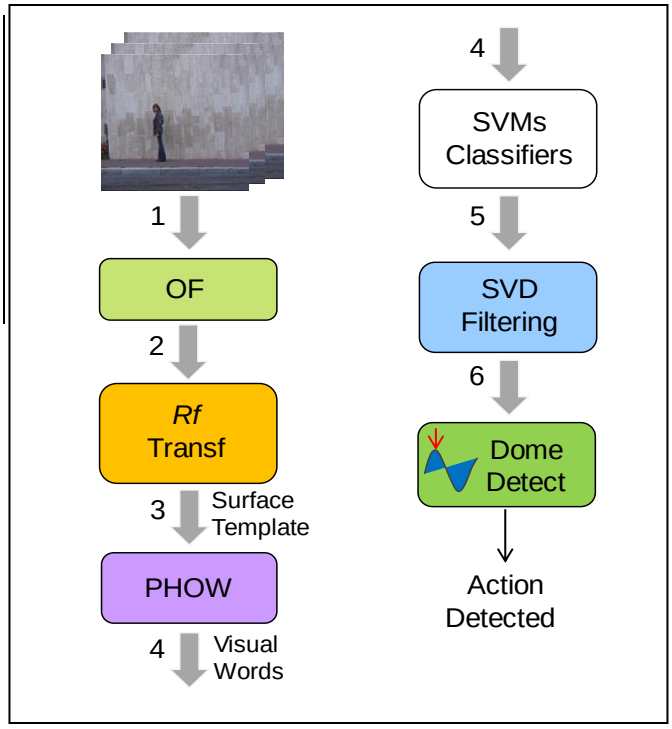

Fig. 2. Bloc diagram of the whole segmentation process.

IV. RESULTS

We propose two different experiments. The first one has the objective of detecting a learned human action among others in a sequence containing different actions. The second experiment has the objective of studying the performance of our approach to discriminate known human actions with respect to other type of motion in images. In this second experiment we do not aim to segment a particular human action, but whether it is a known human action or not.

The algorithms used have been tuned as follows; For the BoW we have tested codewords from 100 to 1100 visual words. For PHOW we have tested distances from 1 to 5 pixels between keypoints , and scales from 2 to 10 for the size of the spatial bins. We obtained the best performance using a 1-pixel distance between keypoints in the dense SIFT grid, a 3 pixels size scale, and a Visual Vocabulary of 900 visual words. We have also tested different sizes for the sliding windows to be used in the training stage, obtaining the optimum performance using 360x25 windows.

In our first experiment we have used the Weizmann dataset [12]. It is a widely used sequence database containing a set of human actions. The sequences have been recorded with static camera and background, there are no occlusions, and only a person is moving in all sequences. They do not present serious illumination changes either. This dataset consists in 10 different actions by 9 different persons. Fig. 3 shows some snapshots of the Weizmann dataset.

Fig. 3. Weizmann human actions. Bend, jack, jump, jump, run, side, skip, walk, wave1, wave2.

Following the same procedure as the experiment reported in [7], for the training and segmentation stages, we have attached all the actions performed by the same person to build a single video sequence. We create longer video sequences by concatenating single-action sequences. These sequences including 10 different actions, are fed into the system. In this experiment, we have used leave-one-out cross validation method to evaluate our approach. We use the sequence consisting in the 10 actions done by a single person for testing, and the sequences of actions done by the remaining 8 persons are used for training. This process is repeated for all 9 persons.

The same experiment has been repeated for the 4 different templates obtained by using the different $\Re_f$ transforms ($\Re$, $\Re_{max}$, $\Re_{mean}$,and $\Re_{dev}$). Table 1 shows rates of correct action segmentations for each different $\Re_f$. As mentioned in section 2, the $\Re_{max}$ transform yields the highest accuracy segmenting single actions from the complete sequences.

TABLE I. Accuracy of action temporal segmentation using different $\Re_f$ transforms.

Transform	***%***
$\Re$	92,2
$\Re_{max}$	**96,6**
$\Re_{mean}$	80,5
$\Re_{dev}$	92,2

Since the projection using $\Re_{max}$ gives the best results, a 96.6% of correct segmentations, we will use this transform hereinafter. We only are aware that [1], and [7] tested their methods using no pre-segmented actions from Weizmann sequences, and they report an 88.9%, and 87.7% recognition rates respectively. Fig. 4 shows the outputs of the 10 SVM, each of them trained for a specific action. Domes in SVM outputs indicate the presence of a concrete action.

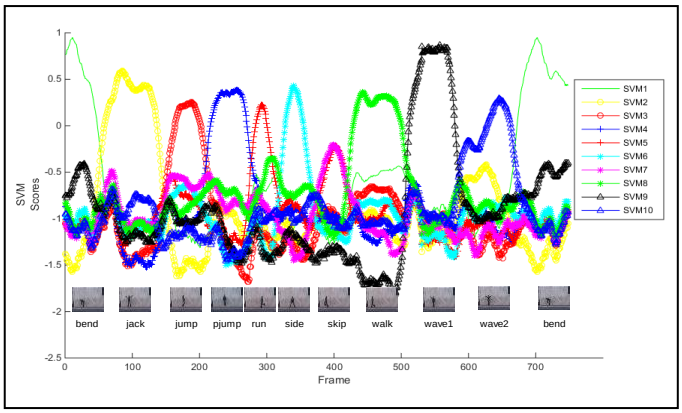

Fig. 4. 10 SVMs scores for a sequence containing 10 actions from Weizmann dataset.

In the experiment described, we are segmenting a concrete action from a set of other human actions taken from the same dataset. Due to the fact that we did not include a null-class corresponding to non-action sequences, this is an experiment closer to action recognition than to action segmentation. In order to segment human actions in a sequence of different motion activities, we have designed a second experiment. In this second experiment we use the Hollywood dataset. This dataset consists in 430 videos including short sequences from 32 movies. The resolution varies from 300*200 to 400*300 depending on the videos. They are filmed with a non-static camera, and the sequences contain cluttered background and occlusions between persons. Figure 5 shows some actions taken from this dataset.

Fig. 5. Some frames from the Hollywood dataset.

For the second experiment, we have labelled all training sequences of the Weizmann dataset as actions and all training sequences of the Hollywood dataset as no-actions. We have merged in single sequences actions of the Weizmann dataset and fragments of videos sequences of the Hollywood dataset. Since, all Weizmann actions and all Hollywood actions are labelled as actions and no-actions respectively, we have not concatenated two Weizmann actions or two Hollywood actions following each other, because the system would detect them as a single action.

Similar to the first experiment, we have used the same leave-one-out cross validation scheme. We have considered a positive score of the SVM as a Hollywood motion and a negative score as a Weizmann human action. Fig. 6 shows the results obtained in this experiment.

This figure shows that our method discriminates very well between Weizmann actions and Hollywood scenes. We have obtained a 100% rate segmenting the learned actions from other movements. Unfortunately, we are not aware of any similar experiments to establish a fair comparative.

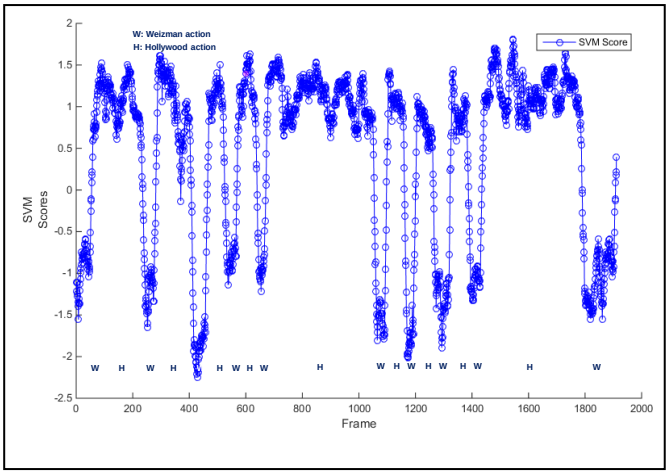

Fig. 6. Example of SVMs scores for a sequence of alternating Weizmann actions and Hollywood scenes.

V. CONCLUSION

In this paper we have proposed a novel technique for temporal segmentation of human actions with video sequences. It is based on the $\Re_f$ transform that projects a sequence into a single *template*. PHOW spatio-temporal descriptors are extracted from these templates and fed to SVM classifiers in order to determine the limits of each human action in a sequence.

A first experiment has shown that our technique overcomes many State Of Art techniques that used non pre-segmented actions on the Weizmann dataset. The second experiment has shown that our technique discriminates very well between Weizmann and Hollywood actions, obtaining a 100% recognition rate.

The results of the first experiment show that our technique could also be used for action recognition. In order to recognize each concrete action, we could have considered the class corresponding to the SVM whose output presents the highest dome. We can report the results of previous works about action recognition using the Weizmann dataset. For example, Gorelick et al. [12] reported the recognition result of 97.8%, Scovanner et al. [17] 84,2%, Klaeser et al. [18] 84,3%, Niebles et al. [19] 90%, Jhuang et al. [20] 98,8%, Vishwakarma et al.[21] 96.64%, and Goudelis et al. [22] 93.4. Unfortunately, their results and ours are not directly comparable. These authors test their approaches with pre-

segmented actions, therefore we can't make a straight comparison of their results with the ones shown in our experiments.

ACKNOWLEDGMENT

This work was supported by the Spanish Ministry of Science and Innovation, project DPI2016-78957- R, and AEROARMS European project H2020-ICT-2014-1-644271

REFERENCES

[1] R. Filipovych and E. Ribeiro, "Learning Human Motion Models from Unsegmented Videos," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-7, 2008.

[2] Richard Souvenir and Kyle Parrigan. Viewpoint manifolds for action recognition. EURASIP Journal on Image and Video Processing, 2009:13 pages, September 2009.

[3] Ying Wang, Kaiqi Huang, Tieniu Tan Human Activity Recognition Based on ℜ Transform Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit., pp. 3722–3729, 2007

[4] Zhu, Pengfei, Hu, Weiming, Li, Li and Wei, Qingdi. "Human Activity Recognition Based on RTransform and Fourier Mellin Transform.." Paper presented at the meeting of the ISVC (2), 2009.

[5] On the determination of functions from their integral values along certain manifolds." IEEE transactions on medical imaging 5.4 (1986): 170-176

[6] S. R. Fanello, I. Gori, G. Metta, F. Odone "Keep It Simple And Sparse: Real-Time Action Recognition" Journal of Machine Learning Research (JMLR), 2013.

[7] M. Hoai, Z.Z. Lan, and F. De la Torre. Joint segmentation and classification of human actions in video. In IEEE CVPR, pages 3265–3272, 2011.

[8] A. Bosch, A. Zisserman, and X. Munoz. "Image classification using random forests and ferns". In International Conference on Computer Vision, 2007.

[9] L. Shao, L. Ji, Y. Liu and J. Zhang, "Human Action Segmentation and Recognition via Motion and Shape Analysis", Pattern Recognition Letters, vol. 33, no. 4, pp. 438-445, Mar. 2012.

[10] Arodz, T., "Invariant object recognition using radon-based transform," Computers and Artificial Intelligence 24(2), 183–199 , 2005.

[11] S. Tabbone, L.Wendling, and J.-P. Salmon. A new shape descriptor defined on the radon transform. Comput. Vis. Image Underst., 102(1):42–51, 2006.

[12] Gorelick, L.; Blank, M.; Shechtman, E.; Irani, M. & Basri, R. "Actions as Space-Time Shapes", IEEE Trans. Pattern Anal. Mach. Intell. 29 (12) , 2247-2253, 2007.

[13] C. Sch¨uldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In ICPR, Cambridge, UK, 2004.

[14] Karlsson, S. M. & Bigün, J, "Lip-motion events analysis and lip segmentation using optical flow", in 'CVPR Workshops' , IEEE, , pp. 138-145, 2012.

[15] A. Bobick and J. Davis, "The recognition of human movement using temporal templates", IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 23, no. 3, pp. 257-267, 2001.

[16] Q. Shi, L. Wang, L. Cheng, and A. Smola. Discriminative human action segmentation and recognition using semi markov model. InComputer Vision and Pattern Recognition,2008

[17] Scovanner, P., Ali, S., Shah, M. "A 3-dimensional sift descriptor and its application to action recognition", ACM Multimedia, pp. 357-360, 2007

[18] Klaeser, A., Marszalek, M., Schmid, C. "A spatio-temporal descriptor based on 3d-gradients", BMVC, pp. 995-1004, 2008

[19] Niebles, J. C.; Wang, H. & Li, F.-F. "Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words", International Journal of Computer Vision 79 (3) , 299-318 , 2008.

[20] H Jhuang, T Serre, L Wolf, T Poggio. "A biologically inspired system for action recognition Computer Vision", IEEE 11th International Conference on, 1-8, 2007.

[21] D. K. Vishwakarma, Ashish Dhiman, Rockey Maheshwari, Rajiv Kapoor "Human Motion Analysis by Fusion of Silhouette Orientation and Shape Features", Procedia Computer Science 57, 438–447, 2015

[22] Goudelis, G., Karpouzis, K., Kollias, S. "Exploring trace transform for robust human action recognition". Patt. Recogn. **46**, 3238–3248, 2013

[23] C. Yuan, X. Li, W. Hu, H. Ling, S. Maybank. "3D R transform on spatio-temporal interest points for action recognition" CVPR (2013), pp. 724–730