Efficient Human Action Recognition Interface for Augmented and Virtual Reality Applications Based on Binary Descriptor

Abassin Sourou Fangbemi[1], Bin Liu[2](✉), Neng Hai Yu[2],
and Yanxiang Zhang[3]

[1] School of Software Engineering, University of Science and Technology
of China, Hefei, Anhui, China
abassino@mail.ustc.edu.cn
[2] School of Information Science and Technology,
University of Science and Technology of China, Hefei, Anhui, China
{flowice,ynh}@ustc.edu.cn
[3] School of Humanities and Social Science,
University of Science and Technology of China, Hefei, Anhui, China
petrel@ustc.edu.cn

Abstract. In the fields of Augmented Reality (AR) and Virtual Reality (VR), Human-Computer Interaction (HCI) is an important component that allows the user to interact with its virtual environment. Though different approaches are adopted to meet the requirements of individual applications, the development of efficient, non-obtrusive and fast HCI interfaces is still a challenge. In this paper, we propose a new AR and VR interaction interface based on Human Action Recognition (HAR) with a new binary motion descriptor that can efficiently describe and recognize different actions in videos. The descriptor is computed by comparing the changes in the texture of a patch centered on a detected keypoint to each of a set of patches compactly surrounding the central patch. Experimental results on the Weizmann and KTH datasets show the advantage of our method over the current-state-of-the-art spatio-temporal descriptor in term of a good tradeoff among accuracy, speed, and memory consumption.

Keywords: Augmented Reality · Virtual Reality · Interaction
Human Action Recognition · Binary motion descriptor
Proximity patches · Real-time

1 Introduction

During the past recent years, many efforts have been devoted to developing more natural and better AR and VR interaction interfaces. Most popular interaction mediums include mobile devices, controllers, keyboard, mouse, fiducial AR makers, body worn-sensor, RGB and depth cameras… As part of the process of scene understanding, Human Action Recognition (HAR) can also be used as an interaction interface. However, though many researches have been conducted in recent years to develop efficient HAR frameworks, their integration in AR and VR is still a challenging task.

L. T. De Paolis and P. Bourdot (Eds.): AVR 2018, LNCS 10850, pp. 252–260, 2018.
https://doi.org/10.1007/978-3-319-95270-3_21

Indeed, the integration of HAR systems in real-time application requires them to be not only accurate but also fast and use a small amount of memory especially for memory limited devices such as mobile phones. In this paper, we address the challenging task of developing an HAR-based interaction interface for AR and VR applications, which is accurate, fast and does not require a lot of memory by exploring a novel method to represent dynamic features and that can efficiently describe motion in a video while achieving a good tradeoff among accuracy, speed, and memory. To do so, we introduce in this paper the following contributions: (1) we propose a new patch-based pattern for motion description in a video, namely the Proximity Patch pattern. In contrast to the previously used patterns, PP uses a compact structure to ensure that all pixel information surrounding a keypoint are used to compute the descriptor of the keypoint. (2) We introduce a new motion descriptor algorithm based on the PP pattern, namely the Binary Proximity Patches Ensemble Motion (BPPEM) descriptor that computes the change in texture at the same point from two consecutive frames using the PP pattern, in contrast to previous works who compute the descriptors using different positions. (3) We propose an extended version of BPPEM (eBPPEM) as a small size and fast motion descriptor using three consecutive frames and that achieves a competitive accuracy on the Weizmann and KTH datasets.

The remaining of the paper is organized as follow. In section two, we perform a literature review of existing HAR systems as an interaction interface for AR and VR applications and of existing binary motion descriptors. In section three, we describe the PP pattern and the BPPEM descriptor' algorithm followed by the analysis of the experiments conducted to evaluate the performance of the descriptor together with the performance analysis of its extension (eBPPEM) and its comparison with the state-of-the-art binary motion descriptors. We conclude in section five with different considerations for future works.

2 Literature Review

2.1 Human Action Recognition for AR and VR Applications

In the quest of developing less cumbersome interaction interface for AR and VR application, gestures and actions recognition systems have been used to capture and analyze user's motion with a camera for interaction purposes. With the introduction of depth cameras such as the Microsoft Kinect, it has become even easier to capture and use human body joints data in AR and VR systems [1, 2]. Though action representation using skeletal data generated by depth sensor is common in AR and VR applications, such approach requires a correct positioning of the depth camera at a certain distance from the user in order to be able to successfully generate the body skeletal data [3]. Additionally, the data generated by depth sensors are easily affected by noise, occlusion, and illumination that can make action representation and recognition more difficult.

Instead of using depth or their corresponding skeletal data to represent actions, RGB videos have proven to be a better alternative in addition to the extensive research already done on RGB images and video data for pose and action recognition. That is the case of the authors in [4] who proposed a real-time system that robustly tracks multiple persons in virtual environments and recognizes their actions through image sequences acquired from a single fixed camera. Unlike [4] which requires some preprocessing on the input data such as shadows and highlights removal in order to obtain a more accurate object silhouette and increase the computational cost, the action recognition method presented in this paper does not require any additional preprocessing step. Though, extensive researches have already been conducted to develop efficient HAR systems from RGB videos, very few as the ones mentioned above are dedicated to their application in AR and VR environment that require not only accurate performance but also fast and real-time performance.

2.2 Patch Based Binary Motion Descriptors

To develop HAR systems that can have possible applications in AR and VR environments, such systems should be simultaneously accurate, fast and use a limited amount of memory if they have to be integrated on low memory devices (mobile AR and VR). To developer faster HAR systems for real-time applications, many research have been focused on binary motion descriptors. In [5], Yeffet et al. proposed a patch-based self-similarity computational approach to characterize changes in motion from one frame to another. Their Local Trinary Patterns (LTP) combines the effective description of LBP with the flexibility and appearance invariance of patch matching methods. The Motion Interchange Patterns (MIP) [6] captures local changes in motion directions between three consecutive frames and includes a motion compensation component, which makes it more robust even in an unconstrained environment. In [7], Whiten and Bilodeau proposed a binary variation of MIP with a great improvement in speed. Recently, Baumann et al. [8] introduced a new and efficient spatio-temporal binary descriptor, namely the Motion Binary Patterns that combines the benefits of the volume local binary pattern (VLBP) and optical flow.

In the works introduced above, the neighborhood patches of the pattern used to compute the descriptors are positioned at a certain pixel distance from the patch centered on the keypoint, leaving out some pixels. This can result in a loss of key information. Additionally, the squared root differences is often used as the similarity metric but it is not rotation invariant. Hence, aside of selecting a more compact pattern for the neighboring patches, our algorithm also uses the Frobenius Norm, which is invariant to rotation as a similarity metric.

3 Overview of the Proximity Patches Pattern and the BPPEM Motion Descriptor

3.1 The Proximity Patches Pattern

In previous literature, to compute binary motion descriptors using a patch-based approach [5–8], a 3 × 3 central patch centered at the coordinates of a detected moving keypoint is compared with a set of 3 × 3 surrounding patches, each positioned at a certain pixel distance from the central patch. Given a moving keypoint between a current and a next frame, the comparison of patches is carried out based on the assumption that the moving patch (central patch) from the current frame may be positioned at the location of one of the surrounding patches in the next frame. Through the computational process of a binary motion descriptor using such pattern, it is obvious that some pixel data surrounding the central frame are left out of the computation since there is a pixel distance gap between the central patch and each of the surrounding patches. In contrast to such exploded structure, we decided to investigate the impact a compact pattern structure can have on the performance of the descriptor. To do so, we propose the Proximity Patches (PP) pattern (Fig. 1(a)), a new patch-based pattern that has a set of surrounding patches compactly positioned around the central patch in order to ensure that all pixel information is included in the computation of the descriptor.

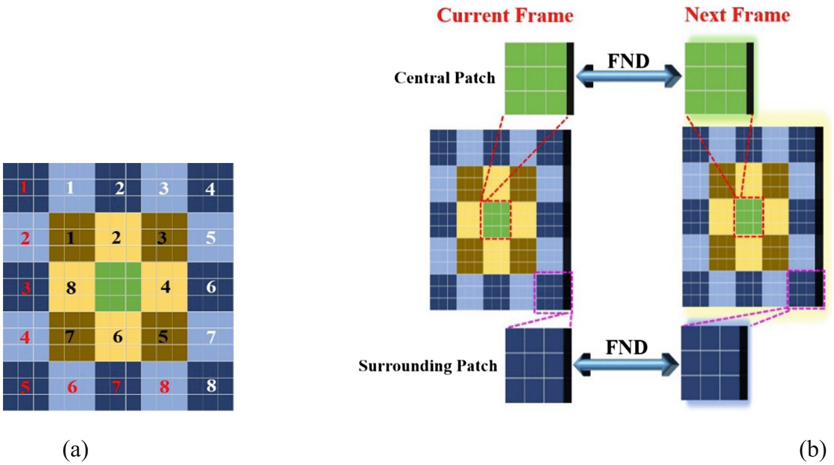
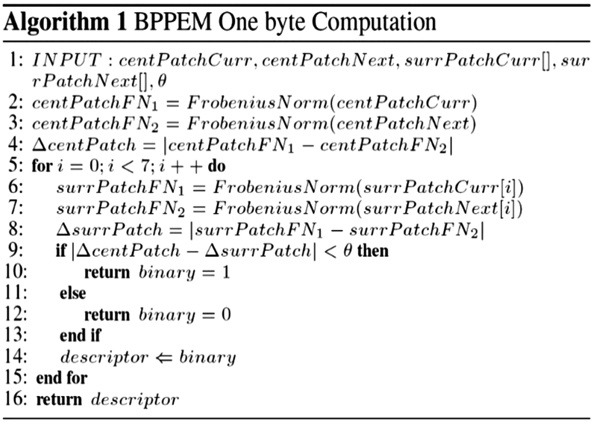


Fig. 1. The Proximity Patches (PP) pattern structure (a) and the computation of BPPEM (b).

3.2 The Binary Proximity Patches Ensemble Motion (BPPEM) Descriptor

BPPEM is a binary motion descriptor that focuses on the change in texture of a patch at the same location on two consecutive frames (Fig. 1(b)) in contrast to other methods that compute the descriptor at different locations on two consecutive frames. The computation of the descriptor is describe in Algorithm 1 and is as follows. After detecting a set of keypoints in a current frame with the SURF keypoint detector, BPPEM computes the Frobenius norms of the central patch in the current frame and the next frame; we then compute their difference as the variation in intensity (ΔcentPatch) of the central patch using Eq. 1. Similarly, we also compute the variation in intensity of each of the surrounding patches between the two consecutive (ΔsurrPatch). Knowing the variations of the central patch and of a surrounding patch, we can estimate the similarity in motion between both patches by computing the difference in their variation. If the difference in their variation is smaller than a predefined threshold θ, we return 1 to symbolize that both patches have a similar variation in intensity, otherwise, we return 0. By comparing the variation of each of the surrounding patches with the one of the central patch, we obtain a binary string of 24 bits grouped in three sets of 8 bits that are concatenated as a 3 bytes descriptor.

$$\Delta(A,B) = \left| ||A||_F - ||B||_F \right| = \left| \sqrt{\sum\nolimits_{i=1}^{m} \sum\nolimits_{j=1}^{n} \left|a_{ij}\right|^2} - \sqrt{\sum\nolimits_{i=1}^{m} \sum\nolimits_{j=1}^{n} \left|b_{ij}\right|^2} \right| \tag{1}$$



Algorithm 1 BPPEM One byte Computation

```
$INPUT : centPatchCurr, centPatchNext, surrPatchCurr[], surrPatchNext[], \theta$
$centPatchFN_1 = FrobeniusNorm(centPatchCurr)$
$centPatchFN_2 = FrobeniusNorm(centPatchNext)$
$\Delta centPatch = |centPatchFN_1 - centPatchFN_2|$
for $i = 0; i < 7; i++$ do
    $surrPatchFN_1 = FrobeniusNorm(surrPatchCurr[i])$
    $surrPatchFN_2 = FrobeniusNorm(surrPatchNext[i])$
    $\Delta surrPatch = |surrPatchFN_1 - surrPatchFN_2|$
    if $|\Delta centPatch - \Delta surrPatch| < \theta$ then
        return $binary = 1$
    else
        return $binary = 0$
    end if
    $descriptor \Leftarrow binary$
end for
return $descriptor$
```

4 Experimental Results and Analysis

4.1 Experimentation Setup: Framework, Datasets, Evaluation Metrics

In order to evaluate the performance of our method, we adopted the bag-of-words representation together with an SVM classifier approach. After detecting keypoints of interest using the SURF detector and encoding them with the BPPEM descriptor, we generate a codebook by picking descriptors randomly and then compute the histogram distribution of each video to train an SVM model with the histogram intersection kernel. During the testing phase, we match each descriptor to the nearest codeword by Hamming distance and follow the same scheme to form k-dimension BoW histogram.

The performance of the descriptors was evaluate on two popular HAR datasets, namely the Weizmann [9] and KTH [10] datasets. The Weizmann dataset contains 10 human actions performed by nine persons and the KTH dataset contains six types of human actions performed by 25 subjects.

Three main metrics were used to evaluate the performance of the system and consist of the accuracy (average classification results with the leave-one-out SVM method), the speed in frame per-second computed during a recognition task from keypoint retrieval until classification, and the size of the descriptor.

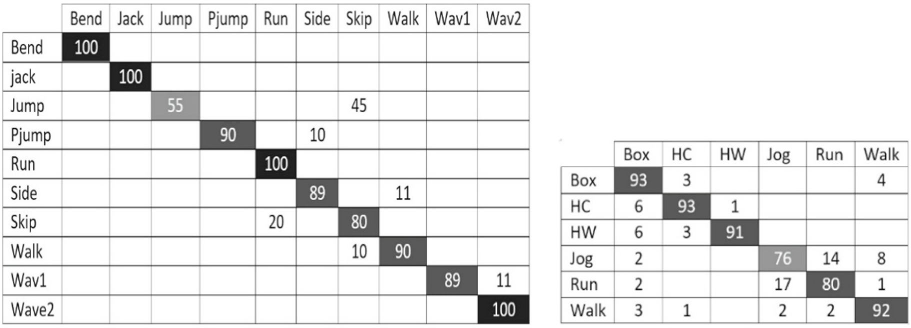
4.2 Experimental Analysis

BPPEM

Table 1 shows experimental results of BPPEM and eBPPEM on the Weizmann and KTH. From the confusion matrix of the performance of BPPEM on both datasets, BPPEM (Fig. 2(a)) has more difficulty in differentiating between the Jump and Skip actions but also between the Jump and Run actions on the Weizmann dataset. In the former case, it is normal that BPPEM performs poorly because of the high similarly between the both actions with the main difference being that the Jump is performed on two legs while the Skip is performed on one leg. Similarly, on the KTH (Fig. 2(b)) dataset, BPPEM has more difficulty in recognizing the Jogging action, which is mostly confused with the Run and Walk actions, with the difference between both actions residing mainly in their speed of execution.

Table 1. Performance of BPPEM and eBPPEM on the Weizmann and KTH datasets

	Size (Bytes)	Weizmann		KTH	
		Accuracy (%)	Speed (fps)	Accuracy (%)	Speed (fps)
BPPEM	3	89.72	56.68	87.10	54.25
eBPPEM	6	92.22	50.64	91.14	46.28



	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wav1	Wav2
Bend	100									
jack		100								
Jump			55				45			
Pjump				90		10				
Run					100					
Side						89		11		
Skip					20		80			
Walk							10	90		
Wav1									89	11
Wave2										100

(a) BPPEM on Weizmann

	Box	HC	HW	Jog	Run	Walk
Box	93	3				4
HC	6	93	1			
HW	6	3	91			
Jog	2			76	14	8
Run	2			17	80	1
Walk	3	1		2	2	92

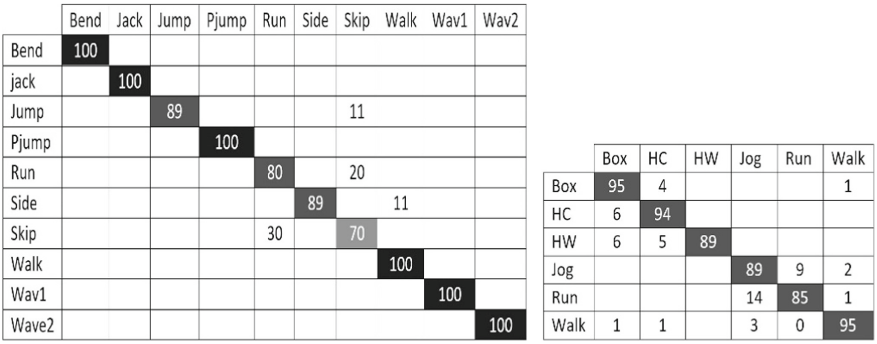
(b) BPPEM on KTH

Fig. 2. Confusion matrix of BPPEM on the Weizmann (a) and KTH (b) datasets

eBPPEM

In the experiments performed above, we computed BPPEM by comparing patches from two consecutive frames whereas an alternative approach as done in [5, 6] consists of computing the descriptor using three consecutive frames. Hence, we also explore the impact the computation of BPPEM from three consecutive frames has on the performance of the descriptor by comparing patches from the current frame and the previous frames and patches from the current frame and the next frame. This result in an extended version of BPPEM (eBPPEM) with a twice increase in size (6 bytes).

Experimental results in Table 1 shows that the computation of the descriptor using three consecutive frames yields indeed to an improvement of the accuracy on both datasets. Figure 3 shows the confusion matrix of the performance of eBPPEM on the Weizmann (a) and KTH (b) datasets respectively, with eBPPEM being able to differentiate better between similar actions than BPPEM.



	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wav1	Wav2
Bend	100									
jack		100								
Jump			89				11			
Pjump				100						
Run					80		20			
Side						89		11		
Skip					30		70			
Walk								100		
Wav1									100	
Wave2										100

(a) eBPPEM on Weizmann

	Box	HC	HW	Jog	Run	Walk
Box	95	4				1
HC	6	94				
HW	6	5	89			
Jog				89	9	2
Run				14	85	1
Walk	1	1		3	0	95

(b) eBPPEM on KTH

Fig. 3. Confusion matrix of eBPPEM on the Weizmann (a) and KTH (b) datasets

Comparison with the State-of-the-Art Spatio-Temporal Binary Descriptors

From the analysis performed above, it is clear that with eBPPEM, we are able to achieve a good tradeoff among accuracy, speed, and memory on both Weizmann and KTH dataset. Compared to the state-of-the-art accuracy, it can be seen that in contrast to previous works, our method does not require any preprocessing step as done in [8] for example. Moreover, our descriptor is totally a binary descriptor and focus on motion description whereas some of the state-of-the-art methods combined their motion descriptors to an appearance descriptor or other floating-pointing motion descriptors to achieve the highest accuracy. That is the case of [11] who combined the advantages of Hu invariant moments global descriptors with their local binary pattern descriptor to increase performance on the KTH dataset. Though the performance of our descriptor is still competitive to the current state-of-the-art spatio-temporal binary descriptor, it has the additional advantage of being faster and smaller than previous descriptors. Given a video data, the speed computed in frame per second (fps) corresponds to the number of frames that are processed per second during a recognition task, including the keypoint retrieval, feature extraction, vector quantization, bag-of-words and classification substeps. Experimental results show that eBPPEM can performs at 50.64 fps on the Weizmann dataset (Table 2) and 46.28 fps on the KTH dataset (Table 3), fast enough to be used for real-time HAR-based AR and VR applications. Though the use of an SVM classifier has yielded to satisfying results, it is not very suitable for a multi-class problem, and we can achieve better accuracy with our descriptor by using a better classifier such as Random Forest or a Decision Tree.

Table 2. Comparison of the performance of eBPPEM with the state-of-the-art binary motion descriptor on the Weizmann dataset

Methods	Classifiers	Accuracy (%)	Speed (fps)
[8] MBP	RF	**100**	N/A
[5] LTP	SVM	**100**	25
[11] DW LBP +Moments	KNN and DT	91.4	N/A
eBPPEM	SVM	92.22	**50.64**

Table 3. Comparison of the performance of eBPPEM with the state-of-the-art binary motion descriptor on the KTH dataset

Methods	Classifiers	Accuracy (%)	Speed (fps)
[11] DW_LBP +Moments	KNN and DT	**96**	N/A
[6] MIP	SVM	93	N/A
[8] MBP	RF	92.13	N/A
[5] LTP	SVM	90.1	25
[7] MoFREAK	SVM	90	42.05
eBPPEM	SVM	91.14	**46.28**

RF: Random Forest, SVM: Support Vector Machine, KNN: K-nearest Neighbors, DT: Decision Tree

5 Conclusion

In augmented and virtual reality environments, the understanding of the scene and the correct interpretation of involved human gestures, actions or activities are key elements to develop natural and efficient interaction interfaces between the user and the virtual world. In this paper, we propose a new vision-based human-action recognition method to efficiently describe motion in a video. In contrast to previous methods that are also based on patches but used an exploded structure, we introduced a new compact patch pattern (PP pattern) that include all pixel information in the closest vicinity of a keypoint to describe its motion between three consecutive frames. The description is performed by computing a new binary motion descriptor based on the PP pattern (eBPPEM) that evaluates the changes in texture at the same pixel position of a patch and its surroundings from three consecutive frames. The proposed method has the advantage of being simultaneously fast, small and able to achieve a competitive accuracy on the Weizmann and KTH datasets, when compared to the current state-of-the-art spatio-temporal binary descriptor. In future works, we will explore the use of classifiers that are more suitable for multi-class problems such as random forest and together with the design and the implementation of an AR or VR application that used our method as interaction interface.

Acknowledgment. This work is supported by the CAS-TWAS Presidents Fellowship, the National Natural Science Foundation of China (Grant No. 61371192), the Key Laboratory Foundation of the Chinese Academy of Sciences (CXJJ-17S044), and the Fundamental Research Funds for the Central Universities (WK2100330002, WK3480000005).

References

1. Khotimah, W.N., Sholikah, R.W., Hariadi, R.R.: Sitting to standing and walking therapy for post-stroke patients using virtual reality system. In: International Conference on Information and Communication Technology and Systems (ICTS), pp. 145–150 (2015)
2. Sieluzycki, C., Kaczmarczyk, P., Sobecki, J., Witkowski, K., Maśliński, J., Cieśliński, W.: Microsoft Kinect as a tool to support training in professional sports: augmented reality application to Tachi-Waza techniques in judo. In: Third European Network Intelligence Conference (ENIC), pp. 153–158 (2016)
3. Tao, G., Archambault, P.S., Levin, M.F.: Evaluation of Kinect skeletal tracking in a virtual reality rehabilitation system for upper limb hemiparesis. In: International Conference on Virtual Rehabilitation (ICVR), pp. 164–165 (2013)
4. Choi, J., Cho, Y.I., Cho, K., Bae, S., Yang, H.S.: A view-based multiple objects tracking and human action recognition for interactive virtual environments. IJVR **7**(3), 71–76 (2008)
5. Yeffet, L., Wolf, L.: Local trinary patterns for human action recognition. In: IEEE 12th International Conference on Computer Vision, pp. 492–497 (2009)
6. Kliper-Gross, O., Gurovich, Y., Hassner, T., Wolf, L.: Motion interchange patterns for action recognition in unconstrained videos. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 256–269. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33783-3_19
7. Whiten, C., Laganiere, R., Bilodeau, G.A.: Efficient action recognition with MoFREAK. In: International Conference on Computer and Robot Vision (CRV), pp. 319–325 (2013)
8. Baumann, F., Ehlers, A., Rosenhahn, B., Liao, J.: Recognizing human actions using novel space-time volume binary patterns. Neurocomputing **173**, 54–63 (2016)
9. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Tenth IEEE International Conference on Computer Vision (ICCV), vol. 2, pp. 1395–1402 (2005)
10. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 3, pp. 32–36 (2004)
11. Al-Berry, M.N., Salem, M.A.M., Ebeid, H.M., Hussein, A.S., Tolba, M.F.: Fusing directional wavelet local binary pattern and moments for human action recognition. IET Comput. Vis. **10**(2), 153–162 (2016)