Camera Movement and Surrounding Scene Appearance as Contextual Features for Action Recognition

Fabian Caba Heilbron[1,2], Ali Thabet[1], Juan Carlos Niebles[2] and Bernard Ghanem[1]

King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia[1]
Universidad del Norte, Barranquilla, Colombia[2]

Abstract. This paper describes a framework for recognizing human actions in videos by incorporating a new set of visual cues that represent the *context* of the action. We develop a weak foreground-background segmentation approach in order to robustly extract not only foreground features that are focused on the actors, but also global camera motion and contextual scene information. Using dense point trajectories, our approach separates and describes the foreground motion from the background, represents the appearance of the extracted static background, and encodes the global camera motion that interestingly is shown to be discriminative for certain action classes. Our experiments on four challenging benchmarks (HMDB51, Hollywood2, Olympic Sports, UCF50) show that our contextual features enable a significant performance improvement over state-of-the-art algorithms.

1 Introduction

Human action recognition is a challenging task for computer vision algorithms due to the large variabilities in video data caused by occlusions, camera motions, actor and scene appearances, among others. A popular current trend in action recognition methods relies on using local video descriptors to represent visual events in videos [1–3]. These features are usually aggregated into a compact representation, namely the bag-of-features (BoF) representation [4]. The advantage of this simple representation is that it avoids difficult pre-processing steps such as motion segmentation and tracking. In the BoF representation, local descriptors are quantized using a pre-computed codebook of visual patterns. This representation combined with discriminative classifiers such as support vector machines (SVM), has been quite successful in recognizing human actions in controlled scenarios [5, 6]. Due to its simplicity, BoF requires the use of strong, robust and informative features, which can be obtained reliably in such simplified scenarios.

However, recent efforts have been made to collect more realistic video datasets (*e.g.* from movies and personal videos uploaded to video sharing websites [7, 8]), which are useful for evaluating human action recognition methods in more natural settings. In fact, these datasets represent a challenge for existing BoF-based

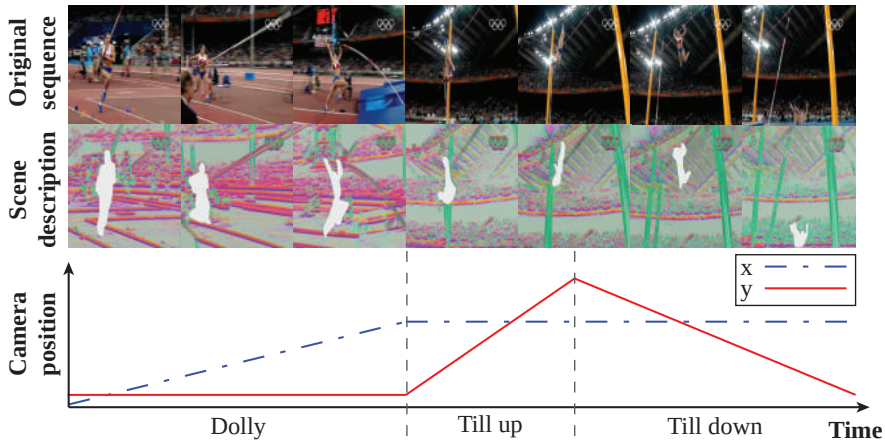

Fig. 1. Some human actions have important correlations with surrounding cues. As observed in the first row, there is a video sequence associated with the human action pole vault. It is also noticeable that the camera moves according to some specific patron for capturing the movement of the subject. Specifically, the camera moves within dolly panning tracking when the athlete is approaching the plant and take off. Then, camera slightly starts to tilling up and tilling down when the person is flying away and falling respectively. Additionally, a better description can be performed if visual appearance of the track field is captured.

methods due to dynamic backgrounds, variations in illumination and viewpoint, and camera motion among other visual nuisances that can severely affect recognition performance. To mitigate the effect of camera motion in describing the action of interest in a video, recent methods [3,9] have proposed using dense point trajectories in a video. In fact, these trajectories can separate background from foreground using a simple camera motion model (*i.e.* an affine or homography transform between consecutive frames). Such separation allows action recognition approaches to robustly extract and describe foreground motion, which is otherwise contaminated by camera motion and the background. Inspired by this work, our proposed method also makes use of these dense trajectories; however, we enlist a more general camera model (by estimating the fundamental matrix between consecutive video frames) that allows for a more reliable separation between foreground and background pixels, especially in non-planar cluttered scenes. Unlike most other methods, we claim that the *context* of a human action, namely global camera motion and static background appearance, can also be used to discriminate between certain human actions. These cues can also be considered as contextual features for an action, which would allow classification algorithms to mine the relationship between the human action and both the background scene as well as the camera motion. The appearance of the scene in which an action occurs can be helpful in recognizing the action, as validated by

previous work in [8]. For example, a 'cooking' action tends to occur indoors, while a 'jogging' action usually exists outdoors. Interestingly, the manner in which the *cameraman* records a particular action can also be indicative of the action. For example, camera zoom with minimal panning usually indicates an action that is spatially limited to a smaller physical space (*e.g.* juggling balls), while significant panning is indicative of actions that require a much larger spatial support (*e.g.* practicing long jump). Our proposed approach mines these two sources of contextual information, as well as, the separated foreground motion to describe and recognize an action. Figure 1 illustrates our claims.

Related work

A large body of work has studied the problem of human action recognition in video. For a survey of this work, we refer the reader to [10]. In this section, we give an overview of previous work that is most relevant to our proposed method.

Action Recognition Pipeline. The majority of action recognition methods rely on local descriptors to represent visual events in videos [1–3]. Traditionally, these features are usually aggregated into a compact representation using the bag-of-features (BoF) framework [4,22]. Moreover, recent studies show that using soft encoding techniques, such as Fisher Vectors [11] and Vectors of Locally Aggregated Descriptors (VLAD) [12], can lead to a boost in action recognition performance. These representations combined with discriminative classifiers such as support vector machines (SVM), have been quite successful in discriminating human action classes. However, as discussed in [13], there remain many details of the overall action recognition pipeline that can be extensively explored, including feature extraction, feature pre-procesing, codebook generation, feature encoding and pooling and normalization. In this paper, we propose a new set of features that can be used to address some of the limitations of conventional feature extraction methods.

Feature Extraction. When applied to videos with substantial camera motion, traditional feature extraction approaches [1,2] tend to generate a large number of features, which are inherently dependent on the camera motion in a video, thus, limiting their discriminative power among action classes. In order to overcome this issue, Wu *et al.* [14] propose the use of Lagrangian point trajectories for action description in videos acquired with moving cameras. Their method compensates for global camera motion and only extracts features that exhibit motion independent of the camera movement, thus, outperforming traditional feature extraction algorithms. Park *et al.* [15] use a weak video stabilization method based on extracting coarse optical flow to isolate limb motion while canceling pedestrian translation and camera motion. Wang *et al.* [3] present a method for action recognition using dense sampling of point trajectories. Their method handles large camera motions by limiting the maximum length of tracked trajectories. Despite their simplicity, these dense trajectory features have been shown

to achieve a significant performance improvement as compared to conventional detection of sparse and salient spatiotemporal features [1].

More recent methods improve upon the aforementioned dense trajectory features. For example, Jain *et al.* [16] propose a method to estimate more reliable trajectory features for action recognition. This method provides additional reliability and robustness to the feature extraction stage by initially decomposing optical flow into dominant and residual motions. Dominant motion is estimated using an affine frame-to-frame motion model and is subtracted from the computed optical flow to obtain the residual motion, which is attributed to the human action of interest. Similarly, 'improved trajectories' are proposed in [9] to stabilize features and compensate for simple camera motion. This is done by fitting a frame-to-frame homography (using RANSAC) to separate moving points of the human action from those of the background. By explicitly canceling out the camera motion, their framework improves the performance of several motion descriptors, including trajectory shape, histogram of optical flow (HOF), and motion boundary histograms (MBH). While these methods have been successful in separating background/residual motions, contextual cues of human actions are usually discarded, thus, ignoring relevant information such as the static scene appearance and distinct camera motions correlated with specific actions.

Moreover, a few approaches have investigated ways to involve background scene information in the action recognition pipeline. Marszalek *et al.* [8] incorporate context information from movie scripts by modeling the relationship between human actions and static scenes based on textual co-occurrence. While such textual co-occurrence helps recognition, they are restricted only to video sources where scripts are available. In [17], multiple feature channels are integrated from different sources of information including human motion, scene information, and objects in the scene. However, this approach makes use of all pixels (corresponding to both the human action and background scene) to generate a global descriptor of the static scene [18]. Rather than computing such a holistic representation, our proposed method separately computes a scene descriptor only from the extracted background, a motion descriptor from the extracted foreground trajectories, and a camera motion descriptor from the estimated transformations between consecutive frames.

In this paper, our goal is to reliably alleviate the effect of camera motion, as well as, incorporate features describing the surrounding of an action to build a richer representation for human actions. We are motivated by the fact that most videos are filmed with an intention and therefore there exists a correlation between the inherent camera motion in a video and the portrayed human action itself. We encode this intention with a weak camera motion model based on frame-to-frame fundamental matrices in a video. To the best of our knowledge, this is the first work to mine such a relationship between human actions and the filming process.

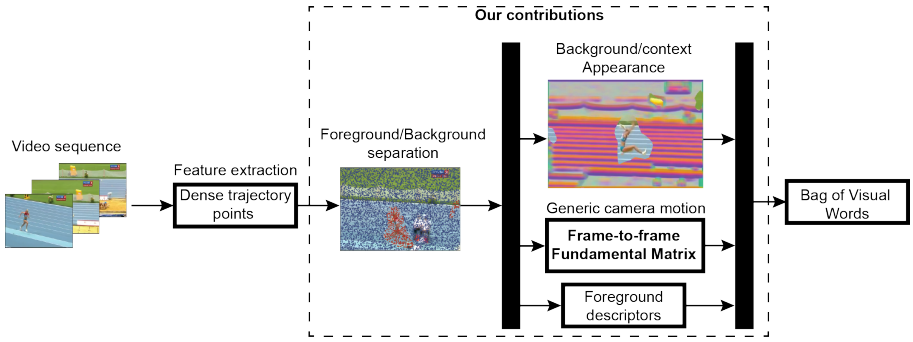

Fig. 2. Given a video sequence, a set of dense points trajectories are extracted. Then, a Fundamental Matrix is used for both applying a camera compensation and separating foreground/background trajectories. Each type of trajectories are encoded by different type of descriptors. Specifically, a low level global motion is used to generally describe the camera movement. Moreover, surrounding scene appearance is explicitly computed on background trajectories. Traditional foreground descriptors (*e.g.* MBH, HOF, HOG and trajectory shape) are also aggregated in actions description. Finally, this set of descriptors are encoded separately using the BoF framework.

2 Proposed Methodology

This section gives a detailed description of our proposed approach for action recognition in video. The methodology in this paper follows the conventional action recognition pipeline. Given a set of labelled videos, a set of features is extracted from each video, represented using visual descriptors, and combined into a single video descriptor, which is used to train a multi-class classifier for recognition.

In this paper, we use dense point trajectories (short tracks of a densely sampled set of pixels in a video [9]) as our primitive features. By estimating frame-to-frame camera motion (fundamental matrix), we separate foreground trajectories corresponding to the action from background ones. Each type of trajectory is represented using a different descriptor. Foreground trajectories are represented using conventional visual properties (*e.g.* MBH, HOF, HOG, and trajectory shape), while the surrounding scene appearance is described using SIFT. Foreground and background trajectories are then encoded separately using the BoF framework as illustrated in Figure 2. Unlike other action recognition methods, we not only use the frame-to-frame camera motion to separate foreground from background, but we also use it to *describe* a video. This is done by encoding all frame-to-frame fundamental matrices in a video using the BoF framework. We use all three descriptors (foreground, surrounding scene appearance, and camera motion) to train a multi-class classifier for recognition. In this paper, we argue and show that combining a foreground-only description [9] with

additional cues (background/context and camera motion) provides a richer and more discriminative description of actions.

2.1 Camera Movement

Since videos are normally filmed with the intention of maintaining the subject within the image frame, there exists a relationship between the estimated camera movement and the underlying action. In this paper, we argue and show that this relationship can be a useful cue for discriminating certain action classes. As observed in the three top rows of Figure 4, there is a correlation between how camera moves and the subject, *e.g.* in the second row, the *cameraman* operates a till down in order to register the movement of the diver. Here, we do not claim that this cue is significant for all types of actions, since very similar camera motion can be shared among classes. Also, several actions not involves subjects translations, as noticed in Figure 4 (**last two rows**). Instead of using a homography to encode camera motion, we estimate the more general fundamental matrix for each pair of frames in a video using the well-known 8-point algorithm [19]. As mentioned earlier, a homography is suitable to describe camera motion when the camera is not translating or when the background is planar; however, it is not applicable in more complex or cluttered scenes.

In our experiments, we calculate **camera motion descriptors** as follows. After estimating all pairwise fundamental matrices using RANSAC, we encode the camera motion of a video using the BoF framework. We call this descriptor CamMotion and it is complementary to other visual descriptors of the video. Unlike most existing work, we embrace camera motion and employ a low-level feature to capture this global motion in the video.

2.2 Surrounding Scene Appearance

We now consider a camera compensation using our global motion model introduced in section 2.1. Exploiting the well performance of this compensation, we can easily separate trajectory points associated with the background, which tend to present a small displacement over the length of the trajectory. Taking advantage of this, we threshold trajectory displacement to obtain a foreground-background separation. The trajectory displacement is computed as follow:

$$D = \sum_{j=t}^{t+L-1} \left((x_{t+1} - x_t)^2, (y_{t+1} - y_t)^2 \right). \quad (1)$$

Trajectory points are associated with the background if $D <= \alpha$. Otherwise, those trajectory points are labeled as foreground. Empirically, we set this threshold value to $\alpha = 3$ pixels.

Figure 3 shows an example of our foreground-background separation in a video associated with the action *long jump*. Here, foreground and background trajectories are color-coded in red and blue respectively. Clearly, the foreground

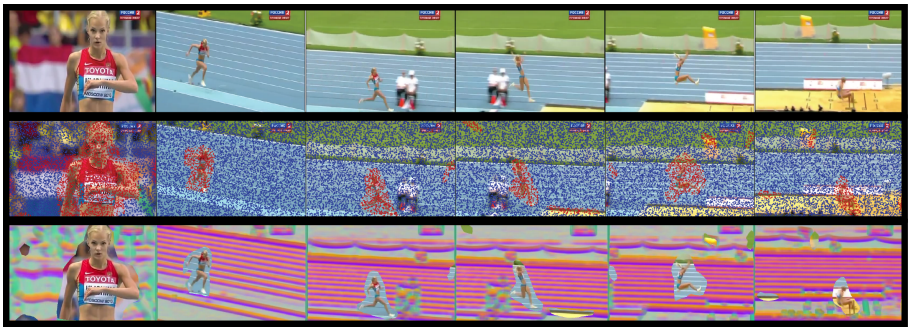

Fig. 3. Obtained results from our foreground-background separation and illustration of the encoded information by the surrounding scene features. **Top**. Frame sequence sampled from a long jump video. Note that camera is panning to follow the subject. **Middle**. Camera compensation allows to perform a background-foreground separation. Noticeably, foreground feature points are mostly related with the subject. **Bottom**. Illustration of information captured by our surrounding SIFT. In order to achieve a meaningful illustration, descriptor dimensionality is reduced to 3 dimensions to produce a color-coded image. As illustrated, surrounding appearance is captured only from pixels related with the scenario *i.e.* avoiding pixels related to the subject that executes the action.

trajectories correspond to the underlying action itself, while background trajectories correspond to *static* background pixels undergoing camera motion only. Our proposed separation will allow each type of trajectory (foreground and background) to be represented independently and thus more reliably than other methods that encode context information using features from the entire frame [8].

In practice, we calculate **foreground descriptors** that consist of Trajectory Shape, HOG, HOF, and MBH computed over dense trajectories as in [9]. In the following section, we detail how surrounding scene appearance is encoded.

2.3 Background/Context Appearance

Human actions could be recognized by a set of cues. Beyond local motion and appearance properties of an action, the surrounding in which an action is performed is a critical component to recognize actions. For example, a 'springboard' action can only be executed if there is a pool, which has distinctive appearance properties. This motivates us to encode the visual appearance of the static scene. Surrounding scene appearance is encoded using SIFT descriptors [20] around trajectory points associated with the background. We detect SIFT keypoints in a dense manner and then filter out those that fall within the union of foreground trajectories. Context appearance focuses more on the scenario itself, as observed in Figure 3. Additionally, in Figure 5 is noticed how context appearance can be used to aggregate meaningful information about the action. For example, all presented thumbnails for the action rowing contains a shared landmark which

Fig. 4. A generic camera motion descriptor can be a useful cue for discriminating specific action categories. As illustrated, the first three rows contains a characteristic correlation between how camera moves and the associated action. Unfortunately, this type of cue its not significant for all type of actions as shown in the last two rows where camera does not move at all.

can be exploited modeling the appearance of background points. Unlike other methods that scene encode context holistically in a video [8], separating the background/context from the foreground produces a more reliable and robust surrounding descriptor.

2.4 Implementation details

Codebook generation. We generate the visual codebook in two different ways: (a) using ***k*-means**, where we cluster the visual space, or, (b) using a **Gaussian Mixture Model (GMM)**, which captures a probability distribution over feature space. In both cases, a codebook is computed for each descriptor separately. Because of trajectory points methodologies produces a large amount of features resulting in intractable codebook computations, it is necessary to sub-sample the features extracted in the training examples for the purpose of codebook generation. In order to establish a trade-off between computation cost and recognition performance, we study the effect of the number of sampled features for computing a visual codebook, as observed in Figure 6. This experiment includes results in two different datasets using k-means to form the visual dictionary. Additionally, we employ a spatial clustering sampling which shows a better performance compared to the uniform sampling. Mentioned spatial clustering finds K centers using k-means over all features in a video. Then, the nearest trajectories on that centers are selected as the features to describe the video. In the following, we

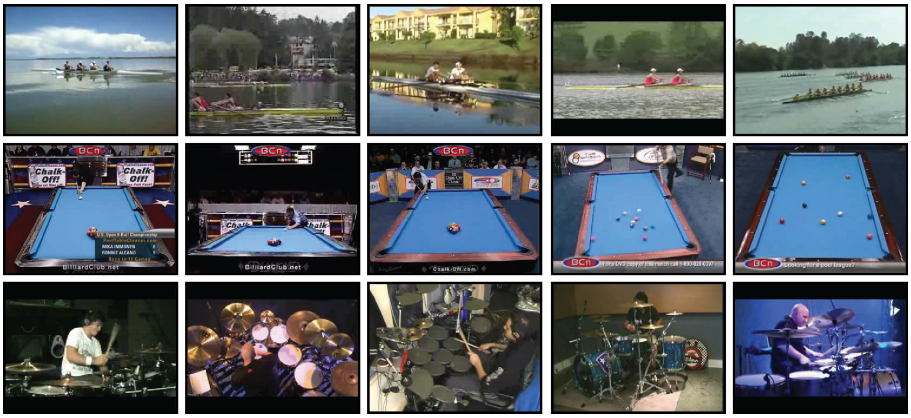

Fig. 5. Each row presents five different thumbnails taken from different videos of UCF50 dataset. **Top** row corresponds to examples of 'rowing'. As observed all thumbnails share distinct background appearance *i.e.* in all water is present and also in the majority there is a common landmark. In the **Middle** row, different billiard examples are depicted. A billiard table and the indoor environment of the action, enable our surrounding appearance descriptor to capture critical information about that action. Finally, **Bottom** row shows examples from the drumming category. Note that these examples share visual cues that are largely ignored if only foreground features are used.

employ approximately 5 millions of feature points (8GB RAM required per descriptor) sampled with a spatial clustering to form visual codebooks.

Feature encoding is performed under two different methodologies: (a) following the traditional histogram quantization (VQ), or, (b) applying the recently introduced Fisher vectors [11]. Different types of **Normalization** are performed to make feature vectors more robust: (a) *l2* normalization (L2) [11], (b) power normalization (PW) [11] and (c) intra-normalization (IN) [13].

Framework representation. We adopt two majors frameworks for action recognition. One of them follows the Bag of Features (BoF) paradigm, using k-means for computing visual codebook, encoding features using VQ and L2 normalization, and finally learning action models with a non-linear SVM with χ^2 kernel within a multichannel approach (MCSVM) as in [21]:

$$K(x_i, x_j) = \exp(-\sum_c \frac{1}{2\Omega_c} D_c(x_i, x_j)), \tag{2}$$

where $D_c(x_i, x_j)$ is the χ^2 distance for channel c, and Ω_c is the average channel distance. Moreover, we implement a more robust framework (Fisher vectors) learning a codebook using a GMM. Consequently, we encode the feature vectors using the Fisher vectors. In this case, we apply three normalization strategies,

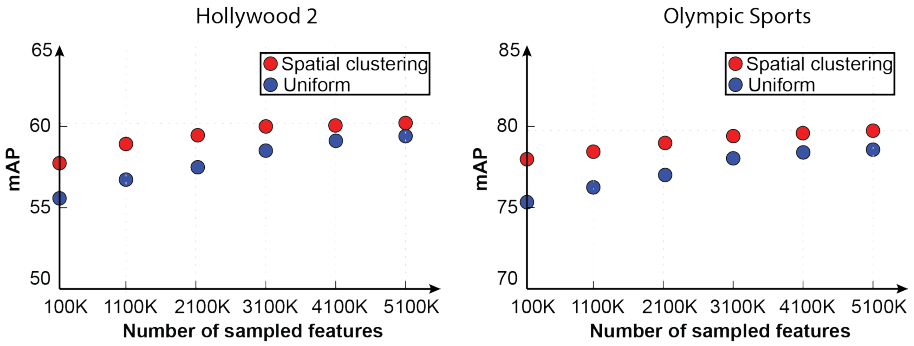

Fig. 6. Due the extensive amount of features generated in a dense trajectory extraction approach, generally sub-sample is required to generate a codebook. Here, we explore the effect of the number of sampled features in the overall performance. Comparison is effectuated on two different datasets under the Bag of Features framework. Additionally, the effect in performance of two different sampling strategies are studied: uniform and spatial clustering. As noticed, selecting more features to form the codebook and using the spatial clustering sampling benefit recognition performance in both datasets evaluated.

L2, PW and IN as proposed in [13]. Finally, normalized channels are concatenated and action models are learned within a linear SVM (LSVM). Described approaches are summarized in Table 1.

Table 1. Comparison of adopted frameworks for action recognition.

Representation ↓	Codebook	Encoding	Normalization	Classifier
Bag of Features	k-means	VQ	L2	MCSVM
Fisher vectors	GMM	Fisher vectors	L2+PW+IN	LSVM

3 Experimental results

3.1 Datasets and evaluation protocol

We use four public datasets [7, 8, 23, 24] and their corresponding evaluation protocols. In this section we briefly describe each dataset.

HMDB51 [7] includes a large collection of human activities categorized on 51 classes. It collects 6766 videos from different media resources *i.e.* digitized movies, public databases and user generated web video data. Due to a large amount of videos contains undesired camera motions, the authors provide a stabilized version of the dataset. However, since we look at the camera motion

as an informative cue, non-stabilized version of the dataset is used. For evaluating performance, we adopt the same protocol proposed by the dataset authors *i.e.* computing the mean accuracy under three fixed train/test splits.

Hollywood2 [8] contains a wide number of videos retrieved from 69 different Hollywood movies. It is divided in 12 categories including short actions such as Kiss, Answer Phone and Stand Up. This dataset remains as one of the most challenging despite the small number of action classes. Change of camera view, camera motion and unchoreographed execution introduces more difficult a the time of recognition. To evaluate performance, we follow the author's protocol where videos are separated in two different sets: a training set of 823 videos and a testing set of 884 videos. We use training videos to learn our action models and then compute the mean average precision (mAP) over all action classes.

Olympic Sports [23] or Olympic comprises a set of 783 sport related YouTube videos. This set of videos are semi-automatically labeled using Amazon Mechanical Turk. This dataset establish new challenges for recognition because of it jumps from simple actions (*e.g.* Kiss) to complex actions (*e.g.* Hammer throw). All of these complex actions are related with olympic sports including actions like *Long jump*, *Pole vault* and *Javelin throw*. As proposed by the author's dataset, we measure performance calculating the mAP over all dataset categories.

UCF50 [24] includes 6618 videos of 50 different human actions. This dataset presents several recognition challenges due to large variations in camera motion, cluttered background, viewpoint, etc. Action categories are grouped into 25 sets, where each set consists of more than 4 action clips. Recognition performance is measured by applying a leave-one-group-out cross-validation and average accuracy over all group splits is reported.

3.2 Impact of contextual features

We conduct further experiments to measure the contribution of our proposed camera movement (CamMotion) and surrounding scene appearance descriptor (SIFT). Our Baseline corresponds to using only Foreground features for describing actions. Per-descriptor performances are compared to that established baseline. Also, we investigate the effect of combining proposed features with Foreground cues. As well, CamMotion and SIFT performance is evaluated under two action recognition representations *i.e.* Bag of Features and Fisher vectors. Below, we present an analysis of obtained results.

Representation. As suggested in recent works [9,11,13] Fisher vectors provides a boosted performance compared to traditional Bag of Feature representations. We found in our experiments that Fisher vectors also boost our contextual descriptors performance, as presented in Table 3. However, we note that using Fisher vectors is less important with our CamMotion descriptor due to its low dimensionality. Even so, Fisher vectors are used for following analysis.

Foreground-background. As described in Section **??**, we perform a weak separation between background and foreground feature points. We measure the effect on performance of this separation in proposed features. We note that this type of weak segmentation provides a significant boost in performance, as observed in Table 2. When feature points are localized on the background, surrounding SIFT focuses on the scene appearance avoiding information of actors and foreground objects. The gain of surrounding SIFT over all the holistic approach is as follow: +0.3% for HMDB51, +4.2% for Hollywood2, +5.2% for Olympics and +3.9% for UCF50. The same behavior is observed with our CamMotion descriptor where performance is boosted in all datasets due to Fundamental Matrix is computed based on background tracks.

Surrounding appearance. While by itself SIFT achieves a discrete performance, it produces notable improvements when combined with foreground descriptors. As Table 3 reports, performance is significantly improved over all datasets. Interestingly, we note that surrounding SIFT produces higher improvements in HMDB51 and UCF50 *i.e.* +2.7% and +2.4% respectively.

Camera movement. Experiment results evidences that action recognition is noticeably improved when a global motion is incorporated to Foreground features. Our CamMotion provides slightly lower contributions in performance than the SIFT descriptor. We observe a significantly contribution over all datasets except on HMDB51 where recognition performance decrease. This negative effect is attributed to the extensive shared shaky camera in video sequences for this dataset. This unable our CamMotion to capture discriminative cues over action categories.

Table 2. Effect of separating background feature points surrounding SIFT and CamMotion. Experimental results consistently show that SIFT exhibit better results when is capturing the surrounding appearance of actions. Conversely, CamMotion and SIFT tend to be more discriminative when computed in non-foreground feature points.

	Feature points		Datasets			
Feature ↓	Foreground	Background	HMDB51	Hollywood2	Olympics	UCF50
SIFT	✓		19.5%	22.1%	33.5%	44.7%
SIFT		✓	**20.1%**	**28.5%**	**39.6%**	**49.8%**
SIFT	✓	✓	19.8%	24.3%	34.4%	45.9%
CamMotion	✓		9.7%	14.9%	19.5%	13.7%
CamMotion		✓	**14.1%**	**22.1%**	**27.2%**	**19.5%**
CamMotion	✓	✓	12.9%	18.7%	21.8%	17.2%

Table 3. Impact of our surrounding scene appearance and camera movement in recognition performance. Bag of Features generally performs poor than Fisher vectors. Both surrounding SIFT and CamMotion show important improvements in performance when they are combined with foreground descriptors.

Features			Datasets			
Foreground	SIFT	CamMotion	HMDB51	Hollywood2	Olympics	UCF50
Framework: Bag of Features						
✓			51.2%	60.1%	79.8%	85.9%
	✓		19.5%	28.7%	36.4%	45.7%
		✓	13.5%	21.8%	26.9%	19.3%
✓	✓		53.8%	60.9%	81.1%	87.2%
✓		✓	50.9%	60.4%	80.6%	86.8%
	✓	✓	20.7%	36.2%	43.7%	50.3%
✓	✓	✓	51.7%	61.6%	81.7%	87.6%
Framework: Fisher vectors						
✓			56.5%	62.4%	90.4%	90.9%
	✓		20.1%	28.5%	39.6%	49.8%
		✓	14.1%	22.1%	27.2%	19.5%
✓	✓		**59.2%**	**63.5%**	**91.6%**	**93.3%**
✓		✓	55.9%	62.9%	91.3%	93.1%
	✓	✓	22.3%	36.5%	46.5%	54.3%
✓	✓	✓	**57.9%**	**64.1%**	**92.5%**	**93.8%**

3.3 Comparison with the state of the art

We set side by side our method with recent methods that address the same application using similar representations, *i.e.* methods that use dense trajectory points to represent video sequences [9, 16, 25] in Table 4. We also present results for our own implementation of [9], which correspond to our baseline (Foreground). The gain over the recent paper [9], which reports the best performance in the literature, is as follow: **+2%** for HMDB51, **+1.4%** for Olympic Sports and **2.6%** for UCF50. We also achieve a comparable performance on Hollywood2 dataset with only 0.2% less in the mAP value. Since Human Detection (HD) is not included in our trajectory extraction stage, a more direct comparison its the non-HD approach of Wang *et al.* [9]. In that case, our method outperforms their improved trajectories in **3.3%** for HMDB51, **1.1%** for Hollywood2, **2.3%** for Olympic Sports and **3.3%** for UCF50.

Acknowledgment. Research reported in this publication was supported by competitive research funding from King Abdullah University of Science and Technology (KAUST). F.C.H. is supported by a COLCIENCIAS Young Scientist and Innovator Fellowship. J.C.N. is supported by a Microsoft Research Faculty Fellowship.

Table 4. Comparison with the state-of-the-art on challenging datasets. Our method improves reported results in the state-of-the-art for three different datasets, HMDB51, Olympic Sports and UCF50 and obtains competitive peformance in Hollywod2.

Approach ↓	HMDB51	Hollywood2	Olympics	UCF50
Jiang *et al.* [25]	40.7%	59.5%	80.6	-
Jain *et al.* [16]	52.1%	62.5%	83.2	-
Wang *et al.* [9] non-HD	55.9%	63.0%	90.2%	90.5%
Wang *et al.* [9] HD	57.2%	**64.3%**	91.1%	91.2%
Our methods with Fisher vectors				
Baseline (Foreground)	56.5%	62.4%	90.4%	90.9%
Foreground + SIFT	**59.2%**	63.5%	**91.6%**	**93.3%**
Foreground + SIFT + CamMotion	**57.9%**	**64.1%**	**92.5%**	**93.8%**

References

1. Laptev, I.: On space-time interest points. IJCV (2005)
2. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. (2005)
3. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: CVPR. (2011)
4. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR. (2008)
5. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: ICCV. (2005)
6. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: ICPR. (2004)
7. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: ICCV. (2011)
8. Marszalek, M., Laptev, I., Schmid, C.: Actions in context. In: CVPR. (2009)
9. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: ICCV. (2013)
10. Aggarwal, J., Ryoo, M.S.: Human activity analysis: A review. ACM Computing Surveys (CSUR) (2011)
11. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: ECCV. (2010)
12. Jégou, H., Perronnin, F., Douze, M., Sánchez, J., Pérez, P., Schmid, C.: Aggregating local image descriptors into compact codes. PAMI (2012)
13. Wang, X., Wang, L., Qiao, Y.: A comparative study of encoding, pooling and normalization methods for action recognition. In: ACCV. (2012)
14. Wu, S., Oreifej, O., Shah, M.: Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In: ICCV. (2011)
15. Park, D., Zitnick, C.L., Ramanan, D., Dollár, P.: Exploring weak stabilization for motion feature extraction. In: CVPR. (2013)
16. Jain, M., Jégou, H., Bouthemy, P.: Better exploiting motion for better action recognition. In: CVPR. (2013)

17. Ikizler-Cinbis, N., Sclaroff, S.: Object, scene and actions: Combining multiple features for human action recognition. In: ECCV. (2010)
18. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. IJCV (2001)
19. Hartley, R.: In defense of the eight-point algorithm. TPAMI (1997) 580–593
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004)
21. Zhang, J., Marszałek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study. IJCV (2007)
22. Niebles, J.C., Escorcia, V.: Spatio-temporal Human-Object Interactions for Action Recognition in Videos In: ICCV. (2013)
23. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: ECCV. (2010)
24. Reddy, K.K., Shah, M.: Recognizing 50 human action categories of web videos. Machine Vision and Applications (2013)
25. Jiang, Y.G., Dai, Q., Xue, X., Liu, W., Ngo, C.W.: Trajectory-based modeling of human actions with motion reference points. In: ECCV. (2012)