This is a postprint version of the following published document:

Naeem, H.B., Murtaza, F., Yousaf, M.H. y Velastin, S.A. (2020). Multiple Batches of Motion History Images (MB-MHIs) for Multi-view Human Action Recognition. *Arabian Journal for Science and Engineering*.

DOI: <https://doi.org/10.1007/s13369-020-04481-y>

Multiple batches of motion history images (MB-MHIs) for multiview human action recognition

Hajra Binte Naeem,^a Fiza Murtaza,^a Muhammad Haroon Yousaf,^{a,b,*} Sergio A. Velastin^{c,d,e}

^aUniversity of Engineering and Technology, Department of Computer Engineering, Taxila, Pakistan, 47080

^bSwarm Robotics Lab, National Centre for Robotics and Automation, Pakistan

^cSchool of Electronic Eng. and Computer Science, Queen Mary University of London, London E1 4NS, UK

^dZebra Technologies Corp., London SE1 9LQ, UK

^eDepartment of Computer Science and Engineering, University Carlos III de Madrid, Colmenarejo, Spain, 28270

Abstract. The recognition of human actions recorded in a multi-camera environment faces the challenging issue of viewpoint variation. Multiview methods employ videos from different views to generate a compact view-invariant representation of human actions. This paper proposes a novel multi-view human action recognition approach that uses multiple low-dimensional temporal templates and a reconstruction-based encoding scheme. The proposed approach is based upon the extraction of multiple 2D Motion History Images (MHIs) of human action videos over non-overlapping temporal windows, constructing Multiple Batches of Motion History Images (MB-MHIs). Then, two kinds of descriptions are computed for these MHIs batches based on (1) a deep Residual Network (ResNet) and (2) Histogram of Oriented Gradients (HOG) to effectively quantify a change in gradient. ResNet descriptions are average pooled at each batch. HOG descriptions are processed independently at each batch to learn a class-based dictionary using a K-Spectral Value Decomposition algorithm (K-SVD). Later, the sparse codes of feature descriptions are obtained using an Orthogonal Matching Pursuit (OMP) approach. These sparse codes are average pooled to extract encoded feature vectors. Then, encoded feature vectors at each batch are fused to form a final view-invariant feature representation. Finally, a linear Support Vector Machine (SVM) classifier is trained for action recognition. Experimental results are given on three versions of a multi-view dataset: MuHAVi-8, MuHAVi-14, and MuHAVi-uncut. The proposed approach shows promising results when tested for a novel camera. Results on deep features indicate that action representation by MB-MHIs is more view-invariant than single MHIs.

Keywords: Human Action Recognition, Motion History Images, Sparse Coding, MuHAVi, Multiview.

* Muhammad Haroon Yousaf, E-mail: haroon.yousaf@uettaxila.edu.pk

1 Introduction

Automated Human Action Recognition (HAR) from videos is one of the challenging tasks in the computer vision domain. Computer vision is a subset of the artificial intelligence domain, making machines visually intelligent. Artificial intelligence techniques are extensively deployed in real time electronics applications. These are used to model a complex system through visual learning and training, where analytical methods usually fail. Developers of security systems are implementing intelligent cameras that are capable of recognizing shapes, lines, edges, and colors, processing this information to enable robust decisions. Major applications of an HAR system are surveillance, content-based analysis, human-computer interaction [1], assisted living, sports monitoring [2], etc. Generally, human actions are defined as the composition of multiple elementary movements of human body parts. Human actions can be categorized as either simple or complex, based on the movement of body limbs, interaction with other humans and objects, the environment and the duration of videos [2].

The recognition of complex human actions has had significant progress in recent times but there are still several challenges [3]. Firstly, there are scale and viewpoint variations introduced by camera motion and position. Secondly, challenges that occur due to environmental variations like lightning conditions, occlusion, noise, and cluttered background [4]. Thirdly, challenges that arise from human anthropometric variations and that result in interclass and intraclass variations [2].

Other challenges include developing a generalized method to recognize any action, recognition speed, the collection of an adequate number of training samples and localizing the action spatially and temporally in video segments [5].

HAR methods that have been proposed, are based on either holistic or local representations of human actions. The former uses global appearance and motion information of an action, whereas the latter is based on local features that could be sparse or dense. Global representations such as motion history images [6], motion energy images [6] or space-time volume [7] depend upon action videos being comprised of human silhouettes. A fine local representation of video requires detection of spatio-temporal interest points (STIPs) [8] that are invariant to scale, viewpoint and illumination conditions. A final description of these local features is obtained by describing either cuboid of pixels around STIPs or trajectories [9] that are formed by tracking STIPs in a sequence of frames. Among these, dense trajectories sampled in optical flow fields have shown state of the art results [10]. The motion information from trajectories can be encoded by edge or motion descriptors, e.g. histogram of oriented gradient (HOG) [11], histogram of optical flow (HOF) [12] and motion boundary histogram (MBH) [13]. In [14], two descriptors (HOF and MBH) are computed over the optical flow to improve recognition accuracy. The optical flow field is rich in motion information but is computationally expensive. Intensity information is encoded by pixel pattern descriptors, such as local binary pattern (LBP) [15] and binary robust independent elementary features (BRIEF) [16]. Such descriptions of pixel patterns are computationally inexpensive but may perform poorly under illumination variations.

Another promising method for action recognition is to learn spatio-temporal features from deep neural networks [17–19]. The top (final) layers of deep networks extract features that depict semantic action information. On the other hand, features from the bottom layers (closer to the inputs) relate to low-level information like edges, curves, and contours. In [17], a generic video descriptor based on a 3D convolutional network was proposed by averaging the outputs of the first fully connected layer of the C3D network. The approach proposed in [18], captures holistic but compact information directly from action videos, where the long-term temporal structure was used to model human action and good practices were introduced to train CNNs, improving action recognition performance. Recently, [19] reports a two-stream CNN to extract spatial (from RGB video) and temporal features (from stacked optical flow) from two separate networks which are then fused to obtain more effective spatiotemporal information.

Significant recognition accuracy has been achieved by approaches discussed above, mainly only when action videos used for testing and training have a common field of view. The performance of these approaches degrades significantly when testing and training samples have a different viewpoint. The major reason for this is dissimilar feature representation of a particular action for different viewpoints. Considering a realistic scenario, such as surveillance where videos are recorded in multi-camera environment, HAR will need to be robust against view variations.

Multi-view action recognition methods have been developed to overcome the problem. In [20, 21], fusion of features from multiple views is carried out for view-invariant representation. A global representation named as multi-view super vector (MVSF) is proposed in [22], composed of relatively independent components derived from a pair of descriptors. However, the computation of MVSF involves matrix multiplication, which might not be affordable in recognition tasks. The issue was addressed in [23], by joint action modeling transferring low-level features into a high-level semantic space. However, these methods enforce the need for all camera views to be available during training.

To address that issue, cross-view methods are preferred. For example, in [24, 25] a view shared feature space is constructed across different views for view-invariant action representation. In [26], a feature space is learned in the frequency domain using space-time template matching. Other cross-view approaches utilize a deep network-based knowledge transfer model [27], by transferring human actions from an unknown view to a shared high-level virtual view. The intuition of such method is to learn feature representation of an unknown view (known as a target view) utilizing data from available views (known as source views), thus handling the problem of missing views. Although these methods provide good recognition accuracy, they require viewpoint information for both source views and target views, which is normally not available in practical applications.

Therefore, many multi-view approaches have been proposed to deal with view-dependent information from the feature representation. These approaches can be divided into two categories i.e. 3D and 2D multi-view. 3D approaches work on the principle of constructing a 3D human model using 2D information in videos [28–31]. On the other hand, 2D approaches completely depend on 2D image data [32–34]. In [35], a feature space is modelled from human silhouette contours. Other 3D approaches incorporate depth information for better reconstruction of the 3D human model [36–38]. Recently, a 3D action recognition approach [39] is proposed by extracting view invariant features from a CNN-based human pose model. 3D information is superior to applying 2D image data from multiple views when it comes to recognition accuracy. However, the algorithms proposed for 3D data reconstruction are computationally complex and might require overlapping multiple camera views [40]. Moreover, depth data can only be computed using RGB-D, which require special sensors for recording videos. Therefore, taking into account realistic constraints, 2D multi-view approaches may be more useful than 3D methods.

The current study is inspired by the investigation of techniques for multi-view action recognition using motion history images (MHIs) in combination with various descriptors and classifiers [34, 41–43]. In [41], multiple MHIs are computed using an overlapping temporal window, to avoid temporal discontinuities. These MHIs are then used on their own to recognize actions using a complex hidden Markov model (HMM). Major drawbacks of this approach are the large number of MHIs per video, the selection of optimal temporal window size and the training of a model without explicitly describing MHIs. The issue of the large number of MHIs per video has been addressed by constructing a multi interval MHI (MMHI) [42] using a fixed number of keyframes and temporal window size. The second drawback has been resolved by computing histogram of oriented gradients (HOG) [34, 44] or histogram of directional derivative (HODD) [43] based description of single MHIs. Later, these descriptions were used to train a simple nearest neighbor (NN) [34, 44] or SVM [43] classifier. This method has demonstrated a remarkable reduction in computational cost with high recognition rates but could not quite handle the view-invariance issue.

Moreover, most of the current approaches exploit encoding schemes based on quantifying the absolute occurrence or absence of a set of visual features. Bag of visual words is a classic example of this approach [45]. Another course of action is evaluating the relative strength of visual features for recognition [46, 47]. In [48], principles of sparse coding are used to aggregate local space-time features, by performing max-pooling on their sparse codes. Recently, the work in [49], has shown a good recognition rate using relative temporal similarity among each action by aggregation of sparse codes, to construct a feature descriptor. Such aggregation method is known as reconstruction-based encoding, which enforces sparse codes to reconstruct the input descriptor [5]. Inspired by the reconstruction property of sparse codes, many view invariant approaches [25, 42,

50, 51] utilize this method to reconstruct descriptors of a missing view. In [42], an attribute matrix that represents a video view as an action attribute was constructed for a high level description. In [50], feature representation, classifier and action specific dictionaries are optimized in parallel for view-specific feature selection. In [51], actions from unseen viewpoints are represented by a view-shared dictionary constructed by optimization of sparse codes. Other methods like [25], fuse sparse representation from multiple dictionaries to ensure view-invariance.

Here we propose a novel approach for human action recognition in a multi-view scenario using multiple batches of motion templates and a reconstruction-based encoding scheme. The proposed approach has four major contributions:

1. Multiple MHIs at various temporal window lengths are efficiently used to discriminate complex human actions among multiple views.
2. A non-overlapping temporal window of fixed length constructs a batch of MHIs. Window size at a later batch is made twice as in earlier batch. Such temporal window selection mitigates the demand of optimal temporal window size and the introduction of a later batch partially avoids temporal discontinuity introduced in an earlier batch as discussed in Sec. 2.1.
3. Instead of quantifying the relative similarity of actions from two views [25], a reconstruction based encoding scheme is employed that effectively quantifies relative temporal similarity among actions from multiple views.
4. Making use of a non-overlapping temporal window, MHIs batches are constructed that are less correlated locally and more correlated globally. Such an MHI construction approach suggests a simple concatenation of MHIs batch features. Hence, a final view-invariant representation is formed by representation level fusion of encoded feature vectors, acquired by average pooling of sparse codes.

The rest of this paper is organized as follows: Section 2 describes the proposed approach in detail. Section 3 presents the experimental evaluation of the proposed method and ablation study to investigate results when handcrafted features are replaced with deep features. Finally, section 4 covers the concluding remarks of this research work.

2 Methodology

This work presents a novel scheme of modeling human actions from multiple views as multiple temporal templates. These templates are a combination of pose and motion variation within a specific time extent, known as motion history images (MHIs). Two kinds of feature descriptions are computed from MHIs batches: handcrafted HOG and deep residual network (ResNet) features. These batches are differentiated by the number of frames required to construct a single MHI. Subsequently, for HOG each batch is processed to learn sparse codes and dictionaries using reconstruction-based encoding in the training phase. In the testing phase, these dictionaries encode sparse codes of test video. Finally, encoded feature vectors are concatenated to acquire view-invariant representation.

Input video sequences that are first processed by segmenting human silhouettes from the background are used. Silhouettes extraction requires segmentation algorithms [52, 53] that effectively differentiate multiple textures in video frames. Despite efficient segmentation algorithm systems, SAR noisy effect resembling speckle [54] is still present. Such noise is reduced through steps introduced in our previous research [34].

A generic block diagram of the complete proposed method is shown in Fig. 1. The following subsections elaborate the proposed approach in detail:

(Fig. 1 to be inserted here)

2.1 Multiple Batches of Motion History Images

MHI is a 2D motion template that records spatio-temporal information of a video sequence in the form of a single grayscale image [6]. This grayscale image represents the variation in motion by smoothly varying lower to higher grayscale values, where the brighter regions in the image show the most recent presence of motion. In the case of human activities, the brightest region shows the most recent pose of a human for that particular action. As an illustration, a few MHIs of human action videos are shown in Fig. 2. Similarly, complex human activities are the combination of multiple distinct poses, which are usually lost in a single MHI. For example, in the MuHAVi dataset, the activity ‘Walk Turn Back’ is composed of two atomic actions: *Walking* and *Turning Back*. By varying the size of the temporal window to compute MHI, we can acquire atomic actions as shown in Fig. 3.

(Fig. 2 and Fig. 3 to be inserted here)

Multiple batches of MHIs (MB-MHIs) are introduced to describe complex human activities. A batch refers to a collection of multiple MHIs. These MHIs are computed using a temporal window of fixed size and sliding it on the complete video in a non-overlapping manner. Frames falling within that window are used to construct a single MHI. Multiple batches imply variation in the size of the temporal window. The strategy to compute MB-MHIs is depicted in Fig. 4.

(Fig. 4 to be inserted here)

The higher the number of MHIs per action, the higher the possibility of retrieving similar poses among different views of the same action. Thus, a feature description becomes less view-dependent. The choice of non-overlapping intervals makes the computation of MHI inexpensive. Here, human poses can also be represented by motion energy images (MEI) as in [6], but a combination of static human pose and motion in MHI maximizes inter-class variation. For instance, actions ‘Walk Turn Back’ and ‘Run Stop’ have quite similar MEIs, but their MHIs are different.

The number of temporal windows varies from video to video, depending upon the duration of the action. Hence, each action video results in a variable number of MHIs. To raise intra-class similarity among MHI requires a constant number of frames i.e. τ per time interval. Also, a fixed temporal window size makes MHI less effective in action speed and camera sampling rate. The algorithm for extracting MB-MHIs is shown in Fig. 5. This is a generic algorithm that computes MB-MHI at M batches for a video with N number of frames. The first *for* loop runs for all batches and the second *for* loop runs for all temporal windows formed at each batch. For a better illustration, the procedure of computing MB-MHIs for two batches is discussed here in detail.

(Fig. 5 to be inserted here)

For an action video V having N number of frames, the starting and ending frames for a non-overlapping window (ω) of size τ at first batch are defined by Eq. (1) in terms of the total number of frames (N).

$$\omega_i = \llbracket i * \tau + 1 \rrbracket \rightarrow \llbracket (i + 1) * \tau \rrbracket \quad s. t. \quad i \in \left\{0, 1, \dots, \frac{N}{\tau} - 1\right\} \quad (1)$$

Hence, for a video V , $\frac{N}{\tau}$ MHIs at first batch are defined as $(H_0^1, H_1^1, H_2^1, \dots, H_{\frac{N}{\tau}-1}^1)$. Here, for a binary frame $I(x, y, t)$, MHI $H(x, y, t)$ at instant t is given by Eq.(2)

$$H(x, y, t) = \begin{cases} \tau & \text{if } I(x, y, t) = 1 \\ \max(0, H(x, y, t - 1) - 1) & \text{otherwise} \end{cases} \quad (2)$$

H_0 is MHI from ω_0 when t is equal to τ . Similarly, MHIs are computed for all temporal windows. In addition to motion transformation within windows $(\omega_0, \omega_1, \omega_2, \dots, \omega_{\frac{N}{\tau}-1})$, these MHIs incorporate the pose of a human at frame $(\tau, 2\tau, 3\tau, \dots, (i + 1) * \tau, \dots, N)$ for a particular action. Moreover, due to non-overlapping windows, $H_1^1, H_2^1, \dots, H_{\frac{N}{\tau}-1}^1$ would have no information of motion variation in H_0^1 . Intuitively, a “temporal discontinuity” occurs between a batch of MHIs. Hence, we have $\frac{N}{\tau} - 1$ discontinuities for a number $\frac{N}{\tau}$ of MHIs.

For the second batch, the size of the temporal window is set to twice the original size to mitigate the influence of such discontinuity. Consequently, the MHIs computed (for the same value of τ) reduce in number to $\frac{N}{2\tau}$ for the second batch, defined as $(H_0^2, H_1^2, H_2^2, \dots, H_{\frac{N}{2\tau}-1}^2)$ as shown in Fig. 4.

To reduce computational complexity, only an odd number of frames are used to compute MHIs at the second batch. The engagement of half of the frames in the MHI computation at the second batch would not result in temporal information loss, as they have already been employed in the first batch.

Comparing MHIs at both batches in Fig. 4, we can observe that H_0^2 depicts a static pose at 2τ , as in H_1^1 , and temporal variation from 1 to 2τ computed by two MHIs (H_0^1 and H_1^1) at the first batch. A similar effect can be seen in all MHIs in the second batch. Therefore, we can say that the proposed MHIs describe action locally at earlier batches and globally at later batches. The abrupt change of gray levels in H_0^2 is due to the skipping of half frames during the computation of MHI. In the second batch, the temporal variation of two MHIs is depicted by a single MHI which recovers half of the temporal discontinuity among $\frac{N}{\tau}$ MHIs in the first batch. For further reduction of discontinuities, a third batch can be constructed. However, in this study, only two batches are considered, as discussed in Sec. 3.3.

2.2 Feature Description and Encoding

Two types of description and encoding schemes are used to quantify MB-MHIs as feature vectors. First is a handcrafted HOG description followed by a reconstruction-based encoding scheme. Secondly, a simple average pooling of deep ResNet features is used. They are discussed in detail in the following sub-sections:

2.2.1 Handcrafted Features

HOG is used initially as a holistic descriptor to describe changes of a gradient in MHI. MHI/HOG description in the action classes is encoded using a set of dictionaries via a reconstruction-based encoding scheme. The proposed feature encoding is performed independently on each batch as shown in Fig. 6. Individual dictionaries are learned from each action class, which are then concatenated to derive a single dictionary per batch. Sparse codes project a set of MHIs from an action video onto class dictionaries. These sparse codes are formed by quantifying the total contribution of atoms in each individual dictionary to the reconstruction of the set of MHIs. In this way, it is possible to quantify the relative similarity between a set of MHIs from each action class. (Fig. 6 to be inserted here)

As an example, consider the case of a dictionary that encodes temporal patterns from one batch. In this case, a set of MHIs from an action class such as ‘Jump Over Fence’ would use for their reconstruction a significant amount of dictionary atoms coming from similar action classes, such

as ‘Jump Over Gap’. Consequently, sparse codes can obtain a feature vector that encodes relative local similarities between the projected set of MHIs and the temporal patterns encoded in each dictionary.

Feature description of MB-MHIs is obtained through the steps shown in Fig. 6. In the first step, holistic representation of MHIs at both batches uses a HOG descriptor [11]. The HOG descriptor translates changes of the gradient within the MHI in the form of histogram bins. The MHI is divided into blocks of uniformly spaced cells, and histograms at each block count the occurrence of gradient magnitudes. These histograms are then concatenated to form a single feature vector presenting a HOG description of MHIs.

In the second step, the HOG feature vectors and sparse coding are used to build dictionaries at each batch. Let, Z_j^c be the set of HOG descriptors extracted from all MHIs occurring at j -th batch in training videos from class c , where $j \in [1, 2]$ $c \in [1, C]$ and C represents the total number of action classes. For an m dimensional HOG descriptor and n number of MHIs obtained from videos for all views of class c at the j th batch, $Z_j^c \in \mathbb{R}^{m \times n}$. Next, the K-spectral value decomposition (K-SVD) algorithm [55] is used to learn a per class dictionary D_j^c and a sparse representation X_j^c for the j -th batch, which solves Eq. (3):

$$\min_{D_j^c, X_j^c} \|Z_j^c - D_j^c X_j^c\|_F^2 \quad s.t. \quad \|X_j^c\|_0 \leq \lambda_1 \quad (3)$$

For k number of atoms in each dictionary, $D_j^c \in \mathbb{R}^{m \times k}$ and $X_j^c \in \mathbb{R}^{k \times n}$. To restrict the number of non-zero values in columns of X_j^c , λ_1 defines a sparsity constraint. Sparse coefficients in X_j^c concentrate on the part of the representation corresponding to a single view. For each action class, a single dictionary is learned for all views unlike [50] that enforces dictionary learning for each view individually. The D_j^c learnt tend to be the best representations of action from all views, due to the flexibility of the K-SVD algorithm to adapt to a given training data. Moreover, this algorithm generalizes the k-means clustering algorithm leading to fast convergence and low training time [55].

In the third step, the final dictionary for each batch is formed by concatenating dictionaries from each action class. Explicitly, the dictionary at the j -th batch is $D_j = [D_j^1 D_j^2 \dots D_j^C]$. To obtain a final description, sparse codes are retrieved at each batch by projecting the HOG description onto D_j , as shown in Fig. 6. Let, z_j^i represent the HOG description of a video v_i . Given that $z_j \in \mathbb{R}^{m \times n_i}$, n_i is the number of MHIs for video v_i at batch j , it is projected onto dictionary D_j using an Orthogonal Matching Pursuit (OMP) technique by solving Eq. (4):

$$\min_{y_j} \|z_j^i - D_j y_j^i\|_F^2 \quad s.t. \quad \|y_j\|_0 \leq \lambda_2 \quad (4)$$

where, $y_j^i \in \mathbb{R}^{K \times n_i}$, K is the total number of atoms in D_j and λ_2 defines the sparsity constraint on y_j . Next, we compute the similarity of each MHI to the atoms corresponding to each class by using an average-pooling operator. This evaluates the contribution of atoms in D_j to the reconstruction of z_j . We have defined an average pooling operator using Eq. (5)

$$\phi_j^i(z_j^i) = \frac{1}{n_i} \sum_{l=1}^{n_i} y_j^i(l) \quad (5)$$

where sparse vectors y_j^i from i -th video (having n_i MHIs) are average pooled to form a descriptor ϕ_j^i . Hence, for M number of video descriptors at batch j , $\Phi_j = [\phi_j^1 \phi_j^2 \dots \phi_j^M] \in \mathbb{R}^{K \times M}$.

The final feature representation is obtained by concatenating descriptors at each batch, defined as $\Psi = [\Phi_1 \Phi_2]$. This is known as representation level fusion. Such feature fusion tends to improve the action recognition task [5], as MHIs at each batch are less correlated locally and more correlated globally. The final feature dimension is the sum of the number of atoms in the dictionary at each batch. The higher dimension of feature representation eases classification by an SVM as discussed in Sec. 3.3.

2.2.2 Deep Features

Recent developments in the field of human action recognition increasingly rely on deep neural networks for efficient video description. Deep convolutional networks describe human action appearance and motion either by a single spatio-temporal network [17] or by two separate spatio-temporal networks [19]. These networks either require RGB frames [17] or motion features like trajectories and optical flow [19] for CNN (Convolutional Neural Network) features description. In this framework, a view-invariant solution in the 2D domain is proposed. Hence, a pretrained network rich in feature representation of 2D images is suitable to obtain descriptions of MB-MHIs.

Deep residual network (ResNet) has boosted the performance of 2D image recognition applications, such as object detection and face recognition. ResNet introduces an extra connection that skips one or more layers, addressing the issue of diminishing backpropagation of gradient in earlier layers of deep networks. *ResNet101* [56] is a deep residual network with 34 parameter layers. A *ResNet101* model pretrained on the 1.28 million images of the ImageNet database with 1000 classes has been employed in this work. This model was trained on input images of size 224x224 with a mini-batch size of 256. The learning rate was selected to 0.1 when starting and is divided by 10 when the error becomes constant for 60×10^4 iterations. Weighting decay and momentum were kept at 0.0001 and 0.9 respectively. Features are extracted from the last fully connected layer of *Resnet101* [56]. Features from multiple MHIs are averaged [17] to form a single feature vector per video at each batch. Then, features from all batches are concatenated and fed to the classifier for action recognition.

2.3 Action Recognition

Before training a classifier for action recognition, feature representation should be normalized to a common scale. Also, normalization is crucial for the descriptor fusion approach, as it requires the final feature representation to be independent of the number of extracted descriptors. Therefore, L2 normalization has been applied before training the classifier. A simple classifier has been used instead of heuristic sparse representation-based scoring [49] or decision functions as proposed in [42]. As suggested in [57], bounding the sparsity i.e. non-zero coefficients, is more meaningful than setting the allowed reconstruction error. A linear Support Vector Machine (SVM) is trained for action recognition. The choice of classifier kernel is linked to the feature normalization procedure. As reported by [5], a linear SVM is suitable for L-2 normalized features. To facilitate multiclass action recognition, one-vs-all SVM has been used.

3 Experimental Evaluation

3.1 Dataset

To verify the view-invariance capability of the proposed feature representation, the proposed approach has been tested on three variants of a publicly available multi-view dataset: MuHAVi-uncut [34], MuHAVi-MAS-14 [58] and MuHAVi-MAS-8 [58]. MuHAVi-MAS-8 is the simplest in these datasets. It contains manually annotated silhouettes videos from 2 cameras, annotated for

8 actions performed by 2 actors. Videos are recorded from the side and diagonal views of the action zone (mounted at an angle of 45°). On the other hand, MuHAVi-MAS-14 contains similar videos, annotated for 14 actions. Here, the direction of action performance is interpreted as two different actions. For instance, action ‘Run’ in MuHAVi-MAS-8 is annotated as ‘Run Left to Right’ and ‘Run Right to Left’ in MuHAVi-MAS-14. Therefore, the increase in the number of actions with only a difference in the acting path increases the complexity of this version of the dataset.

MuHAVi-uncut is the most complex in these datasets. It contains long uncut realistic silhouette videos recorded from 8 cameras for 17 actions and 7 actors. These videos were recorded in a rectangular room from surveillance cameras positioned at 45 degrees. Realistic noisy silhouettes lead to the need for a robust algorithm to overcome the challenges in human action recognition. All these datasets were recorded in a fixed camera setup and ground truths of each actor, action, and camera are available.

3.2 Validation schemes

Results are computed using two cross-validation schemes: Leave one actor out (LOAO) and Leave one camera out (LOCO). LOAO validation scheme is trained on sequences from all actors except one and tested for the remaining actor sequence, in turn. This validation scheme tests robustness against human anthropometric variation. The LOCO validation scheme is trained on sequences from all cameras except one and tested for the remaining camera sequence, in turn. This validation scheme tests the robustness of a proposed method to variations in viewpoint and scale. The algorithm is tested iteratively for each actor and camera as per LOAO and LOCO validation schemes respectively. Average accuracy and standard deviation are then computed to gauge the strength and stability of the proposed algorithm. As this study is mainly concerned to overcome challenges faced due to viewpoint variation, results on LOCO are discussed in detail in Sec. 3.4 while LOAO results are only used for comparison with other methods.

3.3 Parameters selection

Appropriate recognition accuracy depends on the correct selection of parameters. MB-MHIs parameters are tuned to reduce inter-class similarity among MHIs. Two series have been selected and the value of τ is set to 20 for all action classes. As videos are sampled at 30 FPS, approximately one MHI is obtained after every 0.7 sec for the first batch and 1.4 sec for the second batch. While performing an action, an average change in human position is about 0.5 seconds. So, the moderate value of τ is selected to limit the number of identical MHIs. The minimum duration for an action completion is 1.5 sec, leading to two MHIs at first batch and one MHI at the second batch. Hence, two batches are enough to get maximum discriminative MHIs. Moreover, MHIs are resized to 100×100 to mitigate the effect of translation and scale variation.

3.3.1 Varying Temporal Window Size and Number of Batches

Fig. 7 shows the effect of varying the value of τ for each dataset. These results show that a lower value of τ causes the final feature representation to be more view-invariant, because a lower value of τ leads to more temporal patterns and poses, dividing complex activities into atomic actions. Therefore, inter-class similarity among activities having some common atomic actions is reduced. Moreover, concatenation of features at each batch leads to a significant increase in recognition accuracy, as features at a subsequent batch recover the temporal discontinuity that occurred in the previous batch.

(Fig. 7 to be inserted here)

HOG parameters are selected as non-overlapping 8×8 -pixel blocks of four 4×4 -pixel cells, 8 orientation bins histogram and L_2 norm block normalization. The dimension of the histogram for each block is 32. For an MHI of 100×100 , there are 169 blocks resulting in HOG feature vectors of 5408 dimensions. A ResNet feature vector from the last fully connected layer is of dimension 1000.

The number of atoms, k , in D_j^c are tuned for each dataset. The tuning of k for the first batch is shown in Fig. 8. For MuHAVi-uncut, k is 12 and 6 for the first and the second batch, respectively. The number of atoms for the second batch is reduced to half, taking into account the reduction in the number of MHIs. Therefore for 17 action classes, the size of D_1 is 204 (12×17) and D_2 is 104 (6×17). λ_1 is set to 1 for K-SVD in Eq. (3) as the number of atoms per class is low. λ_2 for OMP in Eq. (4), is set to 20 and 10 for D_1 and D_2 , respectively. The value of λ_2 is 10% of the size of D_j . The length of the encoded feature vector at each batch after average pooling is equal to the number of atoms in the dictionary at that batch. After feature fusion, the length of the final feature representation is the sum of the number of atoms in the dictionary at each batch. Hence, feature fusion helps categorize actions more easily without increasing computational cost. (Fig. 8 to be inserted here)

3.4 Results and Discussion

In this section, the computational cost of MB-MHIs and the results of leave-one-camera-out (LOCO) are discussed in detail to assess robustness against viewpoint variation. The approach results in promising accuracies of 83.82%, 63.97% and 61.50% on MuHAVi-8, MuHAVi-14, and MuHAVi-uncut, respectively as shown in table 1. As it can be expected, a decrease in performance is observed with rising dataset complexity.

(Table 1 to be inserted here)

3.4.1 LOCO results on MuHAVi-8

Fig. 9 shows the confusion matrix for MuHAVi-8, with an average recognition accuracy of 83.82%. ‘Guard’ is confused with ‘Kick’ and ‘Punch’. ‘TurnBack’ is confused with ‘StandUp’ and ‘Collapse’. Hence, actions having similar pose and motion variation in either initial or ending frames are confused. However, actions with longer durations (more MHIs) like ‘Collapse’ and ‘Run’ have maximum recognition accuracy. MB-MHIs can efficiently discriminate longer duration actions.

(Fig. 9 to be inserted here)

3.4.2 Results on MuHAVi-14 after training on either side view or diagonal view

MuHAVi-14 comprises of actions that differ by the direction of occurrence. This dataset has shown a promising average accuracy of 63.97% for LOCO. To study the phenomena of view-variance with respect to the direction of motion variation, Fig. 10 shows the confusion matrix when training is performed from a side view (camera 4) and tested for the diagonal view (camera3). Meanwhile, Fig. 11 shows the confusion matrix when the classifier is trained for the diagonal view and tested for front view. In the latter case, average accuracy decreases, because diagonally recorded actions are mostly self-occluded and view-invariant feature representation is harder. Moreover, the direction of motion occurrence is more distinguishable in the former case. For instance, accuracy for ‘Stand Up Left’ is completely confused with ‘Stand Up Right’, when tested for a side view, although the odds of confusing these actions decreases to 50 percent when tested from a diagonal view.

(Fig. 10 and Fig. 11 to be inserted here)

3.4.3 LOCO results on MuHAVi-uncut

From the perspective of viewpoint variation, MuHAVi-uncut is the most complex dataset, as it is composed of videos from eight different viewpoints. The average recognition accuracy on this dataset is 62.51%. Fig. 12 and 13 show confusion matrices for MB-MHIs and single MHI, respectively.

(Fig. 12 and Fig. 13 to be inserted here)

Focusing on the effect of view-variation when MB-MHIs are computed instead of single MHIs:

- Multiple MHIs effectively describe complex activities that are formed from multiple atomic actions. For instance, activities ‘Walk Fall’, ‘Pick Up Throw Object’ and ‘Walk Turn Back’ have shown a noticeable gain in recognition of 33%, 19%, and 17% respectively.
- The introduction of multiple batches decreases inter-class similarity for actions performed at some fixed position. An increase in accuracy of 35% and 21% for actions ‘Draw Graffiti’ and ‘Smash Object’ respectively, can be observed. However, the recognition rate is reduced for actions having the same MHI at the start of the action, as observed for ‘Kick’, ‘Look In Car’ and ‘Punch’.
- MB-MHIs proficiently describe activities possessing some motion repetition across time, reducing effects of view variation, as it can be observed that recognition accuracy for ‘Climb Ladder’ and ‘Crawl On Knees’ increase by 25% and 26% respectively.

Table 1 shows a comparison of ResNet deep features results on a single MHI and MB-MHIs for MuHAVi-uncut using LOCO, LOAO, and 5-fold-cross validation schemes. LOCO results illustrate the view-invariance characteristic of MB-MHIs over a single MHI.

3.4.4 Computational Speed

Section 2.2 mentioned computational complexity reduction, when only an odd number of frames are used to construct later batches of MHIs. Fig. 14 shows the computational speed of computing MB-MHIs in frames per second for varying values of τ . With increasing size of the temporal window i.e. τ , the number of frames used to construct single MHIs increases, which results in an overall reduction in speed for computing a batch of MHIs. However, the rate of constructing MB-MHIs increases approximately two times, when only half number of frames falling within a temporal window are incorporated.

(Fig. 14 to be inserted here)

3.5 Comparisons

Table 2 and Table 3 illustrate a comparison with other multi-view approaches reporting results for MuHAVi datasets. Deep temporal motion descriptor (DTMD) [59] learns deep features from MHI using a backpropagation algorithm, which is used for action recognition through the softmax layer. All other methods are based on handcrafted features. In [58], features are extracted in the Fourier domain followed by computation of mean-power spectra for feature representation. In [60], contour-based features were clustered to extract key poses. An action recognition system was proposed in [61], that operates on box and cloud features from videos. Box features provide speed and shape information, whereas cloud features present compact human pose information based on interest points.

(Table 2 and Table 3 to be inserted here)

In contrast to MB-MHIs, the method in [41] constructs multiple MHIs over a single batch, with an overlapping temporal window of average length 25. Then, these templates are used on their

own for action recognition using a complex hidden Markov model (HMM). In [62], two kinds of feature representations MHI/HOG and MHI/HOOF are used in combination to learn a high dimensional feature space. The proposed method complements the approach in [34, 44] regarding feature representation as MHI/HOG.

Results in [61] are only presented for five action classes of MuHAVi-uncut, ‘CrawlOnKnees’, ‘Punch’, ‘RunStop’, ‘WalkTurnBack’ and ‘WaveArms’. Average recognition accuracy and standard deviation of these action classes are compared in table 2. MB-MHIs accuracy has approximately two times less standard deviation as compared to [61]. Hence, the per-class recognition accuracy of MB-MHIs is more stable. On the whole, better performance can be observed for MB-MHIs/Resnet101 for both view and actor variance.

First, for LOCO MB-MHIs outperforms all other MHI based approaches. MB-MHIs/HOG results in a clear rise of accuracy by 8% and 10% for MuHAVi-8 and MuHAVi-14, respectively. MB-MHIs/Resnet101 causes increases in accuracy of 11%, 5% and 6% for MuHAVi-8, MuHAVi-14, and MuHAVi-uncut, respectively. It is worth mentioning that ResNet outperforms the deep features method DTMD, highlighting view invariance characteristic of MB-MHIs, because ResNet features are average pooled from multiple batches of MHIs, whereas DTMD is learned from single MHIs.

Second, for LOAO MHI/DTMD has outperformed all other methods for MuHAVi-uncut. On the other hand, MuHAVi-8 and MuHAVi-14 MHI/HOG and MHI/HOOF [62] give best performances, indicating the importance of HOOF features for invariance to different actors. MB-MHIs define a change in actor pose in multiple intervals of an action, whereas MHI defines a change in actor pose for a complete action. Actor pose in each interval of action may vary for different actors, but the overall change in motion remains the same. Therefore, MHI performs well for actor variance in comparison to MB-MHIs. It can also be observed that MB-MHIs describe a single video with too many MHIs. Thus, increased similar poses among different actions increase misclassification rates. Additionally, accuracy for actor variance reduces when ResNet101 features are used instead of handcrafted features. These results are to be expected as the network is trained on an object dataset that does not cater to actor style variation. Clearly, networks pretrained on object datasets should be fine-tuned on action video datasets.

Experimentation has been performed on a dataset recorded in an indoor environment. For certain illumination conditions, fog or hazing [63, 64] effects are observed in video frames, resulting in occluded human silhouettes. In future, this issue may be resolved employing efficient dehazing algorithms [65, 66].

4 Conclusion and Future Work

In this study, an innovative method for view-invariant feature representation using the HOG description of multiple batches of MHIs has been proposed. Reconstruction based encoding approach exploits the relative strength of actions among views reducing intraclass similarity of actions. The final feature representation constructed from sparse code is low dimensional. MB-MHIs effectively discriminate complex human actions among various views resulting in good results for novel camera setup. MHI construction using a non-overlapping temporal window results in descriptors that are more correlated globally and less correlated locally. Hence, recognition accuracy improves when features at each batch are concatenated. The results of this method for a novel camera setup point to view invariance characteristics on both simple and complex multi-view datasets, compare well to state-of-the-art approaches. The ablation study conducted using

deep features presents interesting results, suggesting further research on fine-tuning of networks pretrained on 2D image data for human action recognition.

For future work, results on other validation schemes can be improved by introducing a weighting scheme to separate out MHIs that are most discriminant among actions. Instead of MHI/HOG, multiple batches of feature descriptions on RGB videos can be introduced for systems where automatic silhouettes extraction is a problem. Also, efficient dehazing algorithms can be employed to remove occluded silhouettes problem, that occurs due to fog/haze.

Acknowledgments

Muhammad Haroon Yousaf has received funding from Higher Education Commission, Pakistan for Swarm Robotics Lab under National Centre for Robotics and Automation (NCRA). S.A. Velastin is grateful to funding received from the Universidad Carlos III de Madrid, the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 600371, el Ministerio de Economía y Competitividad (COFUND2013-51509) and Banco Santander. The authors also acknowledge support from the Directorate of ASR&TD, University of Engineering and Technology Taxila, Pakistan and of Nvidia Corporation for its donation of GPU equipment.

Figures

Fig. 1 Illustration of the complete proposed methodology as training and testing phase at M batches. Training sequences contain videos from multiple views and the testing sequence is from arbitrary view.

Fig. 2 Sample motion history images for three actions: (a) kick (b) punch (c) wave arms

Fig. 3 (a) single MHI for action ‘Walk Turn Back’ (b) two non-overlapping MHIs using 40 frames per MHI for action ‘Walk Turn Back’, representing two atomic actions ‘Walking’ and ‘Turning Back’ respectively

Fig. 4 Computation of MB-MHIs for a single action video on two batches. The size of the temporal window is τ for the first batch and 2τ for the second batch

Algorithm 1: Finding MB-MHI (H_i^k) at k-th batch for i-th temporal window

Input: Silhouette frames $I(x,y,t)$ of video with N frames and temporal window size τ

Output: H_i^k for all temporal windows and batches

Procedure:

```

for  $k = 1 : M$  do % M is total number of batches
    %we have  $\frac{N}{k*\tau}$  windows at k-th batch
    for  $i = 0 : \frac{N}{k*\tau} - 1$  do
        %define starting and ending frame of temporal window  $\omega_i$ 
         $start\_frame = i * k * \tau + 1$ 
         $end\_frame = (i + 1) * k * \tau$ 

        for  $t = start\_frame : end\_frame$  do
            Find  $H(x,y,t)$  using (2)
        end for

         $H_i^k = H(x,y,t)$  % after above loop when  $t = end\_frame$ 
    end for
end for

```

Fig. 5 Algorithm for extracting MB-MHIs at M number of batches

Fig. 6 Reconstruction based encoding of HOG description of MB-MHIs

Fig. 7 Accuracy variation with the value of tau (τ) and number of batches using LOCO on all datasets

Fig. 8 Performance of MuHAVi-uncut varying the per class number of atoms at the first batch

Accuracy: 83.82%

Collapse	100	0	0	0	0	0	0	0
Guard	6	50	19	16	0	0	6	3
KickRight	0	6	88	0	0	6	0	0
PunchRight	0	0	0	100	0	0	0	0
Run	0	0	0	0	100	0	0	0
Standup	0	0	0	0	0	100	0	0
TurnBack	8	0	0	0	0	8	83	0
Walk	0	0	0	0	13	0	0	88

Collapse Guard KickRight PunchRight Run Standup TurnBack Walk

Fig. 9 Confusion matrix for MuHAVi-8 leave one camera out (LOCO) validation scheme

Accuracy: 64.71%

CollapseLeft	75	0	0	25	0	0	0	0	0	0	0	0	0	0
CollapseRight	100	0	0	0	0	0	0	0	0	0	0	0	0	0
GuardToKick	0	0	50	13	0	0	0	0	0	0	0	38	0	0
GuardToPunch	0	0	38	25	0	0	0	0	0	0	0	38	0	0
KickRight	0	0	0	13	88	0	0	0	0	0	0	0	0	0
PunchRight	0	0	0	0	0	100	0	0	0	0	0	0	0	0
RunLeftToRight	0	0	0	0	0	0	100	0	0	0	0	0	0	0
RunRightToLeft	50	0	0	0	0	0	0	50	0	0	0	0	0	0
StandupLeft	0	0	0	0	0	0	0	0	100	0	0	0	0	0
StandupRight	0	0	0	0	0	0	0	50	50	0	0	0	0	0
TurnBackLeft	0	0	0	0	0	0	0	0	50	50	0	0	0	0
TurnBackRight	0	0	0	0	0	0	0	0	0	0	100	0	0	0
WalkLeftToRight	0	0	0	0	0	0	0	25	0	0	0	75	0	0
WalkRightToLeft	50	0	0	0	0	0	0	0	0	0	0	0	50	0

CollapseLeft CollapseRight GuardToKick GuardToPunch KickRight PunchRight RunLeftToRight RunRightToLeft StandupLeft StandupRight TurnBackLeft TurnBackRight WalkLeftToRight WalkRightToLeft

Fig. 10 Confusion matrix of MuHAVi-14 when side view is used for training and the diagonal view is used for testing

Accuracy: 63.24%

CollapseLeft	75	25	0	0	0	0	0	0	0	0	0	0	0	0
CollapseRight	75	0	0	0	25	0	0	0	0	0	0	0	0	0
GuardToKick	38	0	38	0	0	25	0	0	0	0	0	0	0	0
GuardToPunch	13	0	25	0	13	50	0	0	0	0	0	0	0	0
KickRight	0	0	0	0	100	0	0	0	0	0	0	0	0	0
PunchRight	0	0	0	0	0	100	0	0	0	0	0	0	0	0
RunLeftToRight	0	0	0	0	0	0	100	0	0	0	0	0	0	0
RunRightToLeft	0	0	0	0	0	0	0	75	0	25	0	0	0	0
StandupLeft	0	0	0	0	0	0	0	0	0	100	0	0	0	0
StandupRight	0	0	0	0	0	0	0	0	0	100	0	0	0	0
TurnBackLeft	0	0	0	0	0	0	0	0	0	0	100	0	0	0
TurnBackRight	25	0	0	0	0	0	0	0	0	0	0	75	0	0
WalkLeftToRight	0	0	0	0	0	0	75	0	0	0	0	0	25	0
WalkRightToLeft	0	0	0	0	0	0	0	0	0	0	0	0	0	100

CollapseLeft CollapseRight GuardToKick GuardToPunch KickRight PunchRight RunLeftToRight RunRightToLeft StandupLeft StandupRight TurnBackLeft TurnBackRight WalkLeftToRight WalkRightToLeft

Fig. 11 Confusion matrix of MuHAVi-14 when the diagonal view is used for training and the side view is used for testing

Accuracy: 61.51%

ClimbLadder	49	0	12	0	2	0	2	2	0	8	22	0	0	2	0	0	0
CrawlOnKnees	0	71	0	0	9	3	5	1	1	3	2	3	2	0	0	0	2
DrawGraffiti	2	6	45	2	0	4	2	8	10	10	4	0	0	2	0	0	4
DrunkWalk	2	0	0	66	5	4	5	0	1	2	0	0	1	0	2	13	0
JumpOverFence	0	3	1	0	88	2	5	1	0	0	0	0	2	0	1	0	0
JumpOverGap	1	4	0	3	3	55	5	1	7	3	2	4	2	0	2	4	2
Kick	2	1	0	4	5	9	41	9	4	2	11	2	3	0	3	2	2
LookInCar	1	1	9	1	0	3	7	46	6	8	9	0	2	3	1	4	1
PickupThrowObject	0	1	2	1	1	11	5	4	54	4	6	1	7	2	1	1	0
PullHeavyObject	1	4	2	1	1	0	0	3	2	85	0	0	0	0	0	1	0
Punch	2	2	0	2	1	5	17	8	6	4	37	5	3	1	2	0	3
RunStop	0	2	0	0	0	3	4	4	4	4	2	67	0	1	2	7	0
ShotGunCollapse	0	12	0	0	1	7	5	2	6	1	2	1	61	0	1	1	0
SmashObject	0	2	0	0	8	0	1	0	4	0	1	0	0	82	1	0	1
WalkFall	1	1	0	7	5	2	14	2	6	1	0	9	2	0	41	6	1
WalkTurnBack	0	1	0	2	0	1	1	10	0	5	0	5	0	0	4	71	1
WaveArms	2	2	11	1	0	4	3	0	1	0	1	1	0	2	2	2	68

ClimbLadder CrawlOnKnees DrawGraffiti DrunkWalk JumpOverFence JumpOverGap Kick LookInCar PickupThrowObject PullHeavyObject Punch RunStop ShotGunCollapse SmashObject WalkFall WalkTurnBack WaveArms

Fig. 12 Confusion matrix of MuHAVi-uncut for leave one camera out scheme using MB-MHIs

Fig. 13 Confusion matrix of MuHAVi-uncut for leave one camera out scheme using single MHI

Fig. 14 Computational speed of computing MB-MHIs in frames per second vs. temporal window length (τ). (Orange) Using all the frames falling in a temporal window. (Blue) Using only an odd number of frames falling in the temporal window i.e. half of the frames

Tables

Table 1 Results of ResNet deep features on MuHAVi-uncut

Method	Accuracy (%)		
	LOCO	LOAO	5-Fold-Cross Validation
Single MHI	70.54	80.73	84.30
MB-MHIs	79.23	86.44	91.23

Table 2 Comparison of performance on five action classes of MuHAVi-uncut

Method	LOCO	LOAO

	Accuracy (%)	Standard Deviation	Accuracy (%)	Standard Deviation
Box and Cloud Features [61]	81.43	4.35	97.50	1.60
MB-MHIs/HOG	78.89	2.40	94.87	2.00
MB-MHIs/Resnet101	93.17	2.20	98.00	0.86

Table 3 Comparison of results with other methods

Dataset	Method	LOCO		LOAO	
		Accuracy (%)	Standard Deviation	Accuracy (%)	Standard Deviation
MuHAVi-8	Fourier domain features [58]	50	6	-	-
	Bag of key poses [60]	57.4	4.6	83.08	3.6
	MHIs for overlapping windows + HMM [41]	-	-	85.9	-
	MHI/HOG [44]	75.74	-	92.3	-
	MHI/HOG and MHI/HOOF [62]	-	-	95.7	2.8
	MB-MHIs/HOG	83.82	4	95.58	2.9
	MB-MHIs/Resnet101	86.76	4	85.29	4
MuHAVi-14	Fourier domain features [58]	45.53	6	-	-
	Bag of key poses [60]	50	5.4	75.53	4.3
	MHIs for overlapping windows + HMM [41]	-	-	75	-
	MHI/HOG [44]	54.41	-	81.6	-
	MHI/HOG and MHI/HOOF [62]	-	-	86.93	4.6
	MB-MHIs/HOG	63.97	5	82.35	4
	MB-MHIs/Resnet101	59.55	5.7	73.53	5.3
MuHAVi-uncut	Bag of key poses [60]	31.4	6	56.7	11
	MHIs for overlapping windows + HMM [41]	-	-	83.9	6
	MHI/HOG [34]	54.4	5	84.1	4
	MHI/DTMD [59]	73.29	-	89.66	-
	MB-MHIs/HOG	61.5	3.9	79.8	3.9
	MB-MHIs/Resnet101	79.23	3.4	86.44	3.4

References

1. Efthymiou, N., Koutras, P., Filntisis, P.P., Potamianos, G., Maragos, P.: Multi-view fusion for action recognition in child-robot interaction. 2018 25th IEEE International Conference on Image Processing (ICIP). 455–459 (2018). <https://doi.org/10.1109/ICIP.2018.8451146>

2. Aggarwal, J.K., Ryoo, M.S.: Human activity analysis: A review. *ACM Computing Surveys (CSUR)*. 43, 16 (2011)
3. Sigurdsson, G.A., Russakovsky, O., Gupta, A.: What Actions are Needed for Understanding Human Actions in Videos? *Proceedings of the IEEE International Conference on Computer Vision*. 2017-October, 2156–2165 (2017).
<https://doi.org/10.1109/ICCV.2017.235>
4. Sharifzadeh, F., Akbarizadeh, G., Seifi Kavian, Y.: Ship Classification in SAR Images Using a New Hybrid CNN–MLP Classifier. *Journal of the Indian Society of Remote Sensing*. 47, (2018). <https://doi.org/10.1007/s12524-018-0891-y>
5. Peng, X., Wang, L., Wang, X., Qiao, Y.: Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *Computer Vision and Image Understanding*. 150, 109–125 (2016).
<https://doi.org/https://doi.org/10.1016/j.cviu.2016.03.013>
6. Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis & Machine Intelligence*. 257–267 (2001)
7. Yilmaz, A., Shah, M.: Actions sketch: A novel action representation. In: *IEEE Conference on Computer Vision and Pattern Recognition*. pp. 984–989 (2005)
8. Willems, G., Tuytelaars, T., Van Gool, L.: An efficient dense and scale-invariant spatio-temporal interest point detector. In: *European conference on computer vision*. pp. 650–663 (2008)
9. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: *Proceedings of the IEEE international conference on computer vision*. pp. 3551–3558 (2013)
10. Wang, H., Kläser, A., Schmid, C., Liu, C.-L.: Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*. 103, 60–79 (2013)
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: *international Conference on computer vision & Pattern Recognition (CVPR'05)*. pp. 886–893 (2005)
12. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. pp. 1–8 (2008)
13. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: *European conference on computer vision*. pp. 428–441 (2006)
14. Wang, H., Oneata, D., Verbeek, J., Schmid, C.: A robust and efficient video representation for action recognition. *International Journal of Computer Vision*. 119, 219–238 (2016)
15. Heikkilä, M., Pietikäinen, M., Schmid, C.: Description of interest regions with local binary patterns. *Pattern recognition*. 42, 425–436 (2009)
16. Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., Fua, P.: BRIEF: Computing a local binary descriptor very fast. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 34, 1281–1298 (2012)
17. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning Spatiotemporal Features with 3D Convolutional Networks. (2014).
<https://doi.org/10.1109/CVPR.2014.223>
18. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*. (2018)

19. Liu, Z., Hu, H., Zhang, J.: Spatiotemporal Fusion Networks for Video Action Recognition. *Neural Processing Letters*. (2019). <https://doi.org/10.1007/s11063-018-09972-6>
20. Shao, L., Liu, L., Yu, M.: Kernelized multiview projection for robust action recognition. *International Journal of Computer Vision*. 118, 115–129 (2016)
21. Zhu, F., Shao, L., Lin, M.: Multi-view action recognition using local similarity random forests and sensor fusion. *Pattern recognition letters*. 34, 20–24 (2013)
22. Cai, Z., Wang, L., Peng, X., Qiao, Y.: Multi-view super vector for action recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 596–603 (2014). <https://doi.org/10.1109/CVPR.2014.83>
23. R, J.V.C.I., Hao, T., Wu, D., Wang, Q., Sun, J.: Multi-view representation learning for multi-view action recognition q. *Journal of Visual Communication and Image Representation*. 48, 453–460 (2017). <https://doi.org/10.1016/j.jvcir.2017.01.019>
24. Wang, J., Nie, X.: Cross-view Action Modeling , Learning and Recognition. (2014). <https://doi.org/10.1109/CVPR.2014.339>
25. Zhang, C., Zheng, H., Lai, J.: Cross-View Action Recognition Based on Hierarchical View-Shared Dictionary Learning. *IEEE Access*. 6, 16855--16868 (2018)
26. Ulhaq, A., Yin, X., He, J., Zhang, Y.: On Space-Time Filtering Framework for Matching Human Actions Across Different Viewpoints. *IEEE Transactions on Image Processing*. 27, 1230–1242 (2018). <https://doi.org/10.1109/TIP.2017.2765821>
27. Rahmani, H., Mian, A., Shah, M.: Learning a Deep Model for Human Action Recognition from Novel Viewpoints. 40, 667–681 (2016). <https://doi.org/10.1103/PhysRevD.94.065007>
28. Zhang, B., Yang, Y., Chen, C., Yang, L., Han, J., Shao, L.: Action Recognition Using 3D Histograms of Texture and A Multi-Class Boosting Classifier, (2017)
29. Ershadi-Nasab, S., Noury, E., Kasaei, S., Sanaei, E.: Multiple human 3D pose estimation from multiview images. *Multimedia Tools and Applications*. 77, 15573–15601 (2018)
30. Gu, J., Ding, X., Wang, S.: Action recognition from arbitrary views using 3D-key-pose set. *Frontiers of Electrical and Electronic Engineering*. 7, 224–241 (2012)
31. Zhang, D., Shah, M.: Human pose estimation in videos. In: *Proceedings of the IEEE International Conference on Computer Vision*. pp. 2012–2020 (2015)
32. Sargano, A., Angelov, P., Habib, Z.: Human Action Recognition from Multiple Views Based on View-Invariant Feature Descriptor Using Support Vector Machines. *Applied Sciences*. 6, 309 (2016). <https://doi.org/10.3390/app6100309>
33. Chun, S., Lee, C.: Human action recognition using histogram of motion intensity and direction from multiple views. (2016). <https://doi.org/10.1049/iet-cvi.2015.0233>
34. Murtaza, F., Yousaf, M.H., Velastin, S.A.: Multi-view human action recognition using 2D motion templates based on MHIs and their HOG description. *IET Computer Vision*. 10, 758–767 (2016). <https://doi.org/10.1049/iet-cvi.2015.0416>
35. Gonz, L., Velastin, S.A., Acu, G.: Silhouette-based human action recognition with a multi-class support vector machine. In: *9th International Conference on Pattern Recognition Systems (ICPRS 2018)*. p. 5 (2018)
36. Bui, M., Duong, V., Tai, T., Wang, J.: Depth human action recognition depth based on convolution neural networks and principal component analysis. *2018 25th IEEE International Conference on Image Processing (ICIP)*. 1543–1547 (2018). <https://doi.org/10.1109/ICIP.2018.8451232>

37. Liu, M., Liu, H., Chen, C.: Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*. 68, 346–362 (2017)
38. Wang, K., Zhang, G., Xia, S.: Templateless non-rigid reconstruction and motion tracking with a single RGB-D camera. *IEEE Transactions on Image Processing*. 26, 5966–5979 (2017)
39. Rahmani, H., Mian, A.: 3D action recognition from novel viewpoints. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 1506–1515 (2016)
40. Qureshi, F.Z., Terzopoulos, D.: Surveillance camera scheduling: A virtual vision approach. *Multimedia Systems*. 12, 269–283 (2006)
41. Orrite, C., Rodriguez, M., Herrero, E., Rogez, G., Velastin, S.A.: Automatic segmentation and recognition of human actions in monocular sequences. In: *2014 22nd International Conference on Pattern Recognition*. pp. 4218–4223 (2014)
42. Su, T., Chiang, C., Lai, S.: A Multiattribute Sparse Coding Approach for Action Recognition From a Single. *IEEE Transactions on Circuits and Systems for Video Technology*. 26, 1476–1489 (2016). <https://doi.org/10.1109/TCSVT.2015.2409012>
43. Bhorge, S., Bedase, D.: Multi View Human Action Recognition Using HODD. In: *International Conference on Advances in Computing and Data Sciences*. pp. 499–508 (2018)
44. Murtaza, F., Velastin, S.A.: Multi-view Human Action Recognition using Histograms of Oriented Gradients (HOG) Description of Motion History Images (MHIs). (2015). <https://doi.org/10.1109/FIT.2015.59>
45. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. pp. 604–610 (2005)
46. Parikh, D., Grauman, K.: Relative attributes. In: *2011 International Conference on Computer Vision*. pp. 503–510 (2011)
47. Taibi, F., Akbarizadeh, G., Farshidi, E.: Robust reservoir rock fracture recognition based on a new sparse feature learning and data training method. *Multidimensional Systems and Signal Processing*. 30, 2113–2146 (2019). <https://doi.org/10.1007/s11045-019-00645-8>
48. Zhu, Y., Zhao, X., Fu, Y., Liu, Y.: Sparse coding on local spatial-temporal volumes for human action recognition. In: *Asian Conference on Computer Vision*. pp. 660–671 (2010)
49. Alfaro, A., Mery, D., Soto, A.: Action recognition in video using sparse coding and relative features. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 2688–2697 (2016)
50. Wang, W., Yan, Y., Zhang, L., Hong, R., Sebe, N.: Collaborative sparse coding for multiview action recognition. *IEEE MultiMedia*. 23, 80–87 (2016)
51. Zheng, J., Jiang, Z., Chellappa, R.: Cross-view action recognition via transferable dictionary learning. *IEEE Transactions on Image Processing*. 25, 2542–2556 (2016)
52. Akbarizadeh, G.: A New Statistical-Based Kurtosis Wavelet Energy Feature for Texture Recognition of SAR Images. *IEEE Transactions on Geoscience and Remote Sensing*. 50, 4358–4368 (2012). <https://doi.org/10.1109/TGRS.2012.2194787>
53. Tirandaz, Z., Akbarizadeh, G.: Unsupervised Texture-Based SAR Image Segmentation Using Spectral Regression and Gabor Filter Bank. *Journal of the Indian Society of Remote Sensing*. 44, 177–186 (2016). <https://doi.org/10.1007/s12524-015-0490-0>
54. Samadi, F., Akbarizadeh, G., Kaabi, H.: Change detection in SAR images using deep belief network: a new training approach based on morphological images. *IET Image Processing*. 13, 2255–2264 (2019)

55. Aharon, M., Elad, M., Bruckstein, A., others: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*. 54, 4311 (2006)
56. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 770–778 (2016)
57. Traver, V.J., Serra-Toro, C.: Analysis of single-and dual-dictionary strategies in pedestrian classification. *Pattern Analysis and Applications*. 21, 655–670 (2018)
58. Singh, S., Velastin, S. A., Ragheb, H.: MuHAVi : A Multicamera Human Action Video Dataset for the Evaluation of Action Recognition Methods. *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*. 48–55 (2010). <https://doi.org/10.1109/AVSS.2010.63>
59. Nida, N., Yousaf, M.H., Irtaza, A., Velastin, S.: Deep temporal motion descriptor (DTMD) for human action recognition. *Turkish Journal of Electrical Engineering and Computer Sciences*. (2019). <https://doi.org/10.3906/elk-1907-214>
60. Cheema, S., Eweiwi, A., Thureau, C., Bauckhage, C., Iais, F., Augustin, S.: Action Recognition by Learning Discriminative Key Poses. *Computer Vision Workshops*. 1302–1309 (2011)
61. Chou, K.-P., Prasad, M., Wu, D., Sharma, N., Li, D.-L., Lin, Y.-F., Blumenstein, M., Lin, W.-C., Lin, C.-T.: Robust Feature-Based Automated Multi-View Human Action Recognition System. *IEEE Access*. 6, 15283–15296 (2018)
62. Al-Faris, M., Chiverton, J., Yang, L., Ndzi, D.: Appearance and motion information based human activity recognition. In: *IET 3rd International Conference on Intelligent Signal Processing (ISP 2017)*. pp. 1–6 (2017)
63. Singh, D., Kumar, V.: Comprehensive survey on haze removal techniques. *Multimedia Tools and Applications*. 77, 9595–9620 (2018)
64. Singh, D., Kumar, V.: Defogging of road images using gain coefficient-based trilateral filter. *Journal of Electronic Imaging*. 27, 13004 (2018)
65. Singh, D., Kumar, V.: Dehazing of outdoor images using notch based integral guided filter. *Multimedia Tools and Applications*. 77, 27363–27386 (2018)
66. Singh, D., Kumar, V., Kaur, M.: Single image dehazing using gradient channel prior. *Applied Intelligence*. 1–18 (2019)