# Multi-view Human Action Recognition System Employing 2DPCA

Mohamed A. Naiel
Nile University
6th October, Egypt
mohamed.naiel@nileu.edu.eg

Moataz M. Abdelwahab
Nile University
6th October, Egypt
mabdelwahab@nileuniversity.edu.eg

Motaz El-Saban
Cairo Microsoft Innovation Lab Microsoft Research
Cairo, Egypt
motazel@microsoft.com

## Abstract

*A novel algorithm for view-invariant human action recognition is presented. This approach is based on Two-Dimensional Principal Component Analysis (2DPCA) applied directly on the Motion Energy Image (MEI) or the Motion History Image (MHI) in both the spatial domain and the transform domain. This method reduces the computational complexity by a factor of at least 66, achieving the highest recognition accuracy per camera, while maintaining minimum storage requirements, compared with the most recent reports in the field. Experimental results performed on the Weizmann action and the INIRIA IXMAS datasets confirm the excellent properties of the proposed algorithm, showing its robustness and ability to work with small number of training sequences. The dramatic reduction in computational complexity promotes the use in real time applications.*

## 1. Introduction

View-invariant human action recognition is considered as a challenging problem in the field of computer vision. Recently several reports have been published to address this problem. A survey on view-invariant human motion analysis can be found in [1]. View-invariant approaches can be categorized as 3D model based approaches, and 2D model based approaches. The 3D model based approaches, also known as 3D view-invariant pose representation and estimation approaches, are widely used in human action recognition systems [2]–[7]. However, using 3D poses from multiple calibrated cameras usually require high cost computations due to the large number of parameters involved and the high storage requirement. Further, the recovered 2D poses are often not accurate under the perspective projection. These constraints prevent the use of 3D techniques in applications utilizing single camera systems.

On the other hand, the 2D model based approaches have low computational complexity, but need a large number of training examples to capture multiple poses for the same activity performed at different scales. One solution is to have multi-camera system that can capture the same view by different poses. In 2001 Bobick and Davis [8] introduced a view-based temporal template approach using, the Motion Energy Image (MEI) to indicate the presence of motion, and the Motion History Image (MHI) to be a representation of the order of the motion. In this approach a background subtraction was employed followed by collecting a number of frames of size ($\tau$) to produce MEI or MHI. Given a number of MEIs and MHIs for each view/action a statistical descriptions of these images were computed using moment-based features [9]. To recognize an input movement, a Mahalanobis distance is calculated between the moment description of the input and each of the training examples. In a recent approach [10] a multi-camera human activity recognition system is presented. The algorithm is based on multi-view spatio-temporal histogram features obtained directly from acquired images, further the algorithm is implemented in a distributed architecture and has the view-invariant property.

In 2004 Yang *et al.* [11] proposed the Two Dimensional PCA (2DPCA) technique for facial recognition, which has many advantages over the PCA method. It is simpler for image feature extraction, better in recognition rate and more efficient in computation. However, it is not as efficient as PCA in terms of storage requirements.

In this paper a view-invariant human action recognition algorithm employing parallel structure system is presented. This algorithm is based on the input patterns extracted using the Motion Energy Images (MEI), and the Motion History Images (MHI) [8] employing 2DPCA, and a majority voting scheme is used to decide the corresponding action. Experimental results applied on the Weizmann dataset [12], and the INIRIA IXMAS dataset [13] in the spatial domain and the transform domain confirm the excellent properties of the proposed algorithm compared to the most recent approaches in the field.

This paper is organized as follows: Section 2 introduces the overall system description, demonstrating the proposed algorithm with multi-camera system. Section 3 shows experimental results and analysis obtained by testing the proposed algorithm on two public datasets. Finally, conclusions are presented in section 4.

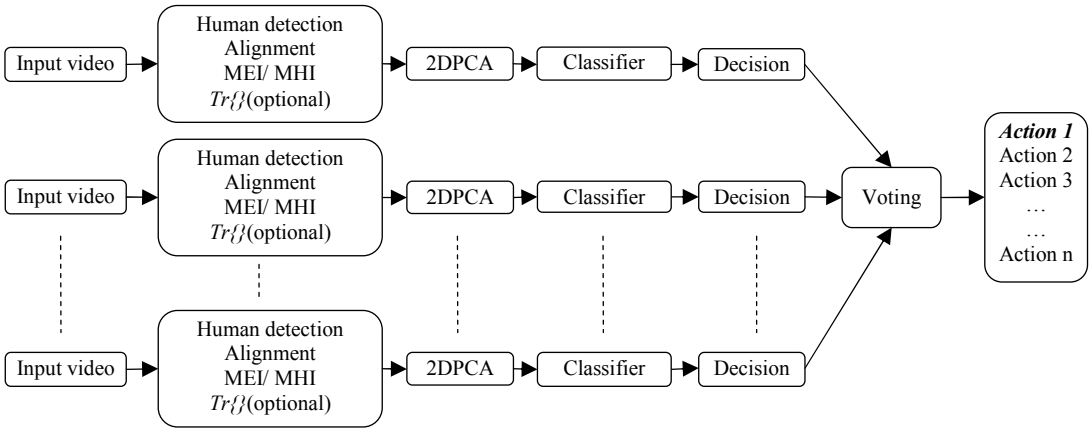



Figure 1: Multi-view human action recognition system.

## 2. Overall System Description

The proposed multi-input/camera human action recognition system, shown in Figure 1, consists of a parallel structure where each path can be considered as an independent human action recognition system that processes every frame as follows. First, *human detection technique* is used to extract clear silhouettes for people. Then a frame *alignment technique* is applied to have the object in the center of every frame. The *MEI* or the *MHI* are used to generate different patterns depending on the input aligned silhouettes. Optionally, a suitable *transform (Tr{})*, i.e. 2D-DCT, can be used to compress the generated patterns from the MEI or the MHI stages. The *2DPCA algorithm* is applied in both training and testing for feature extraction from the input patterns in the spatial-domain or the transform domain. The K- Nearest Neighbor (KNN) *Classifier* is used to infer the most likely class. Finally, a *majority voting* technique is used to decide the corresponding action based on the output of multiple classifiers.

### 2.1. The Proposed Algorithm

Feature extraction from either the raw or transformed MEI/MHI is carried on using 2DPCA. The goal is to deal with the cumulative patterns (MEI, or MHI), in the spatial domain or the transform domain. The algorithm is divided into two modes, the training mode and the testing mode. The following algorithm description is valid for either the spatial or transforms domain MEI and MHI representations.

*Training mode*

In the training mode, videos representing different actions are introduced to the system.

The features of the database are extracted, grouped, and stored as described through the following steps.

1) Read the input MEI or MHI of all the training videos in matrix $M$ of size ($m \times n \times k$), where $m$ and $n$ represent the number of rows and columns for every sequence respectively and $k$ is the size of the all the acceptable training videos.
2) The covariance matrix $S$, of size ($n \times n$), for the $k$ training frames is calculated as follows.

$$S = \frac{1}{k} \sum_{j=1}^{k} (M_j - \overline{A})^T (M_j - \overline{A}) \qquad (1)$$

Where $\overline{A}$ is the mean matrix, of all the $k$ training sequences, of size ($m \times n$).

3) A set of $r$ eigenvectors, $V_q$ of size ($1 \times n$) corresponding to the dominant eigenvalues $\lambda_q$, where $q = \{1, 2, \ldots, r\}$, is obtained for matrix $S$.
4) Store the matrix $V$, where $V = [V_1, V_2, \ldots, V_r]$.
5) Let the matrix $N$, with dimensions ($m \times n$), represent the MEI/MHI of $i^{th}$ training video. Where $i = \{1, 2, \ldots, B\}$, $B$ the maximum number of training videos.
6) Project the matrix $N$ on the matrix $V$ to obtain the feature matrix $F$ of size ($m \times r$).

$$F = NV \qquad (2)$$

7) The feature matrix $F$ is concatenated to produce feature vector (centroid), $C_i = [x_1^{(i)}, x_2^{(i)}, \ldots, x_p^{(i)}]$, the size of this vector is ($1 \times p$), where $p = mr$.
8) Repeat steps 5 to 7 for every video.
9) Store the centroids, $C_i$, $i = \{1, 2, \ldots, B\}$ and their labels, representing each video sequence.
10) Train the suitable classifier using learning technique, i.e. KNN.
11) Repeat steps from 1 to 10 for every input/camera.

*Testing Mode*

In the testing mode the input video is tested according to the following steps:

1) Calculate matrix $N_t$, of size ($m \times n$), where $N_t$ represents the MEI/MHI of the input video sequence in the spatial domain, or the transform domain consistent with the training mode.

2) Repeat steps from 5 to 7 in the training mode, to obtain $C_t$, where $C_t$ represents the centroid of the input action after projection of $N_t$ on $V$.
3) Classification:
   A nearest neighbor classifier is used; the distance between the resulted centroid, $C_t$, and the stored centroids, $C_i$, $i=\{1, 2,..., B\}$ can be measured, using the Euclidean distance (or any distance measure rule) as follows:

$$D_i(C_i, C_t) = \sum_{k=1}^{p} \left\| x_k^{(i)} - x_k^{(t)} \right\|_2 \qquad (3)$$

   Where $\| \|_2$ denotes the Euclidean distance between the two elements $x_k^{(i)}$ and $x_k^{(t)}$. The minimum distance $D_i$ corresponds to the estimated action of the $i^{th}$ video.
4) Repeat steps 1 to 3 for every input/camera.
5) Use a majority voting technique to infer the corresponding action for this view, where don't know decisions are ignored, if the majority voting is not satisfied, the system chooses the decision of the camera with minimum $D_i$.

## 3. Experimental Results and Analysis

The 2D template based approach was first applied on the Weizmann action dataset [12] to measure the performance of the algorithm using a single camera dataset, as shown in section 3.1. The parallel structure algorithm was tested using the IXMAS multi-view dataset [13], as shown in section 3.2.

### 3.1. Weizmann dataset

Four experiments were conducted on the Weizmann action dataset [12] employing the Leave-One-Actor-Out (LOAO) technique, where the 2DPCA is applied on the MEI or the MHI in the spatial domain, or the transform domain. Experimental results were compared to methods that were recently published [15]–[20]. The Weizmann dataset consists of *90* low-resolution (*180 x 144, 50 fps*) video sequences showing nine different actors, each performing *10* natural actions such as walk, run, jump forward, gallop sideways, bend, wave one hand, wave two hands, jump in place, jump-jack, and skip, as shown in Figure. 2. The experiments were applied on the available aligned silhouettes dataset [14] which consists of *90* aligned videos of (*120 x 90, 50 fps*) as shown in Figure 2. The silhouettes contained "leaks" and "intrusions" due to imperfect subtraction, shadows, and color similarities with the background.

In experiments 1, and 2; the recognition system works in the spatial domain, where in experiment 1 the MEI was used, while in experiment 2 the MHI was used. In these experiments, *95%* of the energy of the dominant eigenvalues was maintained.

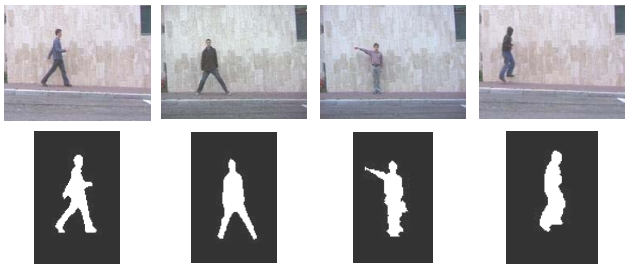



Figure 2: Samples from Weizmann action dataset, where the first row represents the input frame and the second row shows the corresponding aligned silhouette [14].

In experiments 3, and 4; the transform domain 2D-DCT was employed. In experiment 3 the MEI was used where the system maintains on average *86.44%* of the energy in the transform domain, and *99%* of the energy of the dominant eigenvalues. While in experiment 4 the MHI was used where the system maintains on average *99.48%* of the energy in the transform domain, and *99%* of the energy of the dominant eigenvalues.

Table 1 shows the parameters of the four conducted experiments in the training mode where *80* centroids $C_i$ and $i=\{1,..., 80\}$ were generated for every actor from the dataset. In addition to the results obtained in the testing mode in terms of average recognition accuracy, average storage requirement and average testing time. Table 1 shows that the MEI in the transform domain has the highest recognition accuracy *98.89%*, and the lowest average testing time of *17.77* milliseconds, while the MHI in the transform domain has the lowest storage requirement *0.02* Megabytes.

Table 2 compares the average recognition accuracy of the four experiments with the most recent LOAO testing strategies [15]–[18], where higher recognition accuracy is achieved. Experiment 3 has the best accuracy. It worth noting that using the Support Vector Machine (SVM) classifier has not yielded any improvement in accuracy. Table 3 compares the average testing runtime of the four experiments with recent published reports [12, 19, 20], using a Pentium 4, 3.0 GHz, with a Matlab implementation without extra care for optimization. Our proposed method reduced the testing runtime by at least a factor of *70*. Experiment 3 is the best average runtime of *113* milliseconds which is *165* times faster than the best available record [20]. This achievement in the running time (including all steps in the testing mode) is attributed to the simplicity in the testing mode, where it only requires the projection of the MEI/MHI of the tested video on the dominant eigenvectors obtained in the training mode then finding the minimum distance with the stored centroids.

| Parameter/Results | Exp.1, LOAO MEI/SD | Exp.2, LOAO MHI/SD | Exp.3, LOAO MEI/TD | Exp.4, LOAO MHI/TD |
|---|---|---|---|---|
| Dimension of $N$, $M$, $N_t$ | 120x90 | 120x90 | 25x25 | 10x10 |
| Average # of ($r$) | 27 | 26 | 13 | 6 |
| Average size of ($V$) | 90x27 | 90x26 | 25x13 | 10x6 |
| Average size of ($C_i$) | 1x3240 | 1x3120 | 1x325 | 1x60 |
| Average Accuracy | 97.78% | 97.78% | **98.89%** | 97.78% |
| AV. Storage Req. in Mbytes | 1.08 | 1.03 | 0.11 | **0.02** |
| AV. Running Time in msec | 18.37 | 27.40 | **17.77** | 24.93 |
| ST.D. Running Time in msec | 2.10 | 3.418 | **1.89** | 3.23 |

Table 1: Comparison of the average recognition accuracy, the average storage requirement, and the average testing-time on the Weizmann dataset (bold indicates the best performance), where TD= Transform domain, SD= Spatial Domain, AV. = Average, ST.D. = Standard deviation.

| Method | Accuracy | Testing technique |
|---|---|---|
| Exp. 3 | **98.89%** | Leave one-actor out |
| Exp. 1, 2, and 4 | 97.78% | Leave one-actor out |
| Saad & Shah [15] | 95.75% | Leave one-actor out |
| Yang *et al.* [18] | 92.8% | Leave one-actor out |
| Yuan *et al.* [17] | 92.22% | Leave one-actor out |
| Niebles and Fei-Fei [16] | 72.8% | Leave one-actor out |

Table 2: Comparison of the average recognition accuracy on the Weizmann dataset (bold indicates the best performance).

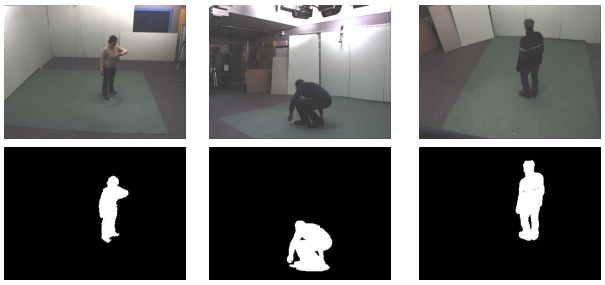
| Method | Average testing runtime | Video size |
|---|---|---|
| Exp. 3 | **113.00 milliseconds** | 144 x 180 x 200 |
| Exp. 1 | 131.25 milliseconds | 144 x 180 x 200 |
| Exp. 2 | 175.49 milliseconds | 144 x 180 x 200 |
| Exp. 4 | 262.95 milliseconds | 144 x 180 x 200 |
| Shah *et al.*[20] | 18.65 seconds | 144 x 180 x 200 |
| Blank *et al.* [12] | 30 seconds | 110 x 70 x 50 |
| Blank *et al.* [19] | 30 minutes | 144 x 180 x 200 |

Table 3: Comparison of the average testing time on the Weizmann dataset (bold indicates the best performance).

## 3.2. IXMAS dataset

The proposed parallel structure algorithm was applied on the extracted silhouettes from IXMAS multi-view dataset [13]. Nine experiments were conducted, four of them are LOAO, and the other five are 6-fold Cross validation. Experimental results were compared to methods that were recently published ([6, 7, 10], and [21] – [24]).

The IXMAS dataset, shown in Figure3, consists of *5* cameras, *13* natural actions, each performed *3* times (also called scenarios) by *12* actors, where the actors are free to change their orientation for each acquisition and there are no particular instructions on how to perform the actions. The resolution of every camera is (*390x291, 23 fps*). The actions are as follows: check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, point, pick up, and throw. To be consistent with most of the available reports we applied our algorithm on *12* actors, *3* scenarios, *11* actions as follows; check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, and pick up. In addition we ignored the top-view camera (camera 5), as the silhouettes are not discriminative.



Check watch Sit down Turn around

Figure 3: Example for different actions, actors and views from IXMAS dataset. First row represents the input frame, second row shows the corresponding silhouette [13].

Most of the available silhouettes have good quality, nonetheless some defects are present which suggested the use of a morphological closing step to enhance the quality of blobs. In addition a frame alignment technique and rescaling to (*161x201*) were applied for every frame.

The experiments can be categorized according to the testing strategy as follows, experiments 5 to 8 are the LOAO cross validation, where the actor is considered with all his *3* scenarios. While experiments from 9 to 13 are the 6-fold cross validation strategy, where every camera is trained separately using ten actors with their three scenarios. Two actors with their *3* scenarios are used in the testing phase. Every experiment was repeated *6* times using different combinations for the training and testing sets.

In experiments 5, 6, 9, and 10; the transform domain 2D-DCT was employed. The MEI in the transform domain was used in experiments 5 and 9, where the system maintains on average *79.5%* of the energy in the transform domain. While in experiments 6 and 10 the MHI in the transform domain was used, where the algorithm maintains on average *99.7%* of the energy in the transform domain. Further, in experiments 5, 6, 9, and 10 the systems maintain *99%* of the energy of the dominant eigenvalues.

In experiments 7, 8, 11, and 12; the MEI or the MHI was used in the spatial domain, where the MEI was used in experiments 7 and 11, while in experiments 8 and 12 the MHI was used. Moreover, the system maintains *90%* of the energy of the dominant eigenvalues.

Table 4 compares our LOAO strategy (experiments from 1 to 4) with the most recent LOAO testing strategies [6, 7, 10, 23], where we achieved the highest recognition

accuracy per camera. Figure 4 shows the confusion matrix for the average overall testing accuracy for experiment 5.

| Method | Actors # | Actions # | Cameras # | Scenarios # | Camera (1) % | Camera (2) % | Camera (3) % | Camera (4) % | Voting using (4) Cameras |
|---|---|---|---|---|---|---|---|---|---|
| Exp. 5 (MEI/TD) | 12 | 11 | 4 | 3 | 78.90 | 78.61 | 80.39 | **77.38** | 84.59 |
| Exp. 6 (MHI/TD) | | | | | **80.35** | **79.82** | 80.11 | 77.08 | 84.59 |
| Exp. 7 (MEI/SD) | | | | | 76.59 | 77.11 | **81.22** | 76.49 | 82.35 |
| Exp. 8 (MHI/SD) | | | | | 75.72 | 76.81 | 79.01 | 75.89 | 82.86 |
| Weinland *et al.* [23] | 10 | 11 | 4 | 3 | N/A | N/A | N/A | N/A | **93.33** |
| Weinland *et al.* [6] | 10 | 11 | 4 | 3 | 65.40 | 70.00 | 54.30 | 66.00 | 81.30 |
| Srivastava *et al.* [10] | 10 | 11 | 4 | 3 | N/A | N/A | N/A | N/A | 81.40 |
| Shah *et al.*[7] | 12 | 11 | 4 | 3 | 72.00 | 53.00 | 68.00 | 63.00 | 78.00 |

Table 4: Comparison of the average recognition accuracy on the IXMAS dataset for LOAO Experiments (bold indicates the best performance), where TD= Transform domain, SD= Spatial Domain, N/A= Not available in published reports.

| | CW | CA | SH | SD | GU | TA | Walk | Wave | Punch | Kick | PU |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CW | **72.22** | 19.44 | 0 | 0 | 0 | 0 | 0 | 8.33 | 0 | 0 | 0 |
| CA | 3.03 | **84.85** | 6.06 | 0 | 0 | 0 | 0 | 6.06 | 0 | 0 | 0 |
| SH | 5.88 | 5.88 | **70.59** | 0 | 0 | 0 | 0 | 17.65 | 0 | 0 | 0 |
| SD | 0 | 0 | 0 | **100** | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| GU | 0 | 0 | 0 | 0 | **94.44** | 0 | 0 | 0 | 0 | 0 | 5.56 |
| TA | 0 | 0 | 0 | 0 | 0 | **94.44** | 5.56 | 0 | 0 | 0 | 0 |
| Walk | 0 | 0 | 0 | 0 | 0 | 2.78 | **97.22** | 0 | 0 | 0 | 0 |
| Wave | 5.56 | 2.78 | 22.22 | 0 | 0 | 0 | 0 | **63.89** | 5.56 | 0 | 0 |
| Punch | 5.56 | 2.78 | 0 | 0 | 0 | 0 | 0 | 5.56 | **80.56** | 0 | 5.56 |
| Kick | 0 | 0 | 0 | 0 | 0 | 0 | 5.56 | 0 | 2.78 | **91.67** | 0 |
| PU | 0 | 2.78 | 5.56 | 0 | 8.33 | 0 | 0 | 0 | 0 | 2.78 | **80.56** |

Figure 4: Confusion matrix for Exp. 5, average accuracy 84.59%, standard deviation 7.01%, where CW=Check watch, CA=Cross arms, SH=Scratch head, SD=Sit down, GU=Get up, TA=Turn around, PU=Pick up.

| Method | Actors # | Actions # | Cameras # | Scenarios # | Cam (1) % | Cam (2) % | Cam (3) % | Cam (4) % | Voting (4) Cameras | Testing technique |
|---|---|---|---|---|---|---|---|---|---|---|
| Exp. 9 (MEI/TD) | 12 | 11 | 4 | 3 | 76.92 | 78.70 | 78.90 | 74.93 | 84.40 | 6-fold CV |
| Exp. 10 (MHI/TD) | | | | | **80.64** | **80.35** | **80.39** | **77.13** | **85.79** | |
| Exp. 11 (MEI/SD) | | | | | 78.03 | 79.77 | 78.93 | 76.26 | 84.83 | |
| Exp. 12 (MHI/SD) | | | | | 73.20 | 78.35 | 78.92 | 75.30 | 81.30 | |
| Liu and Shah [21] | 12 | 13 | 4 | 3 | 76.67 | 73.29 | 71.97 | 72.99 | 82.80 | 6-fold CV |
| | | | | | 72.29 | 61.22 | 64.27 | 70.59 | N/A | LoCO |
| Shah *et al.* [22] | 12 | 13 | 4 | 3 | 69.60 | 69.20 | 62.00 | 65.10 | 72.60 | 41-59 split |
| | | | | | 81.00 | 70.90 | 79.20 | 64.90 | N/A | LoCO |

Table 5: Comparison of the average recognition accuracy on the IXMAS dataset for 6-fold Cross Validation Experiments (bold indicates the best performance), where CV= Cross validation, LoCo= Leave-One-Camera-Out, N/A= Not available in published reports.

Table 5 compares our 6-fold cross validation strategy, experiments from 9 to 12, with the most recent reports [21, 22], where we achieved the highest recognition accuracy per camera *{80.64%, 80.35%, 80.39%,* and *77.13%}* and the best overall accuracy *85.79%.*

| Method | 3D/2D | Run time in msec | | Average # fps |
|---|---|---|---|---|
| | | Average | ST.D. | |
| Exp.5 (MEI/TD) | 2D | **48.69** | 3.80 | **702.67** |
| Exp.6 (MHI/TD) | | 64.76 | 5.56 | 528.28 |
| Exp.7 (MEI/SD) | | 69.14 | 6.14 | 494.84 |
| Exp.8 (MHI/SD) | | 87.68 | 10.03 | 390.22 |
| Exp.9 (MEI/TD) | | 63.43 | 3.46 | 539.38 |
| Exp.10 (MHI/TD) | | 69.17 | 9.91 | 494.64 |
| Exp.11 (MEI/SD) | | 86.70 | 3.83 | 394.60 |
| Exp.12 (MHI/SD) | | 101.23 | **2.66** | 337.98 |
| Weinland *et al.*[6] | 3D | N/A | N/A | 2.5 |
| Lv and R. Nevatia [24] | 2D | N/A | N/A | 5.1 |

Table 6: Comparison of the average testing run time on the IXMAS dataset (bold indicates the best performance), where ST.D. = Standard deviation, N/A= Not available in published reports.

Our reported accuracy compares favorably with most of the previous published reports. However, our biggest gain comes from the computational complexity side. Table 6 illustrates this point and shows that our algorithm runs on at least *337.98* frame/sec, using P4, 3GHz CPU, while the fastest reported algorithms [6], [24] run on *2.5* frame/sec and *5.1* frame/sec respectively, on the same processor. Thus our algorithm is faster than [6] by at least a factor of *135*, and faster than [24] by at least a factor of *66*. This promotes our algorithm to real time applications.

| Method | 3D/2D | Memory req. in Mbytes/Camera | |
|---|---|---|---|
| | | Average | Standard deviation |
| Exp.5 (MEI/TD) | 2D | 0.65 | 0.07 |
| Exp.6 (MHI/TD) | | 0.65 | 0.33 |
| Exp.7 (MEI/SD) | | 5.46 | 0.46 |
| Exp.8 (MHI/SD) | | 5.24 | 0.46 |
| Exp.9 (MEI/TD) | | 0.59 | 0.07 |
| Exp.10 (MHI/TD) | | 0.58 | 0.07 |
| Exp.11 (MEI/SD) | | 4.94 | 0.41 |
| Exp.12 (MHI/SD) | | 4.78 | 0.42 |
| Exp.13 (MEI/TD) | | **0.31** | **0.04** |
| Weinland *et al.*[6] | 3D | 1.72 | N/A |
| Srivastava *et al.* [10] | 2D | 0.32 | N/A |

Table 7: Comparison of the average storage requirement per camera on the IXMAS dataset (bold indicates the best performance), where N/A= Not available in published reports

Table 7 shows that the transform domain experiments achieved a comparable storage requirement per camera to the best records in recent reports [6, 10]. It is worth mentioning that we achieved the minimum storage requirement in experiment 13, by using MEI in the transform domain, where the energy of the dominant eigenvalues was reduced to *90%* instead of *99%*. This reduction led to an accuracy of *82.3%* which is still better than the one reported in [10].

## 4. Conclusions

A view-invariant human action recognition algorithm based on 2DPCA in the spatial domain and the transform domain is presented. This method reduced the computational complexity by at least a factor of *66*, while achieving the highest recognition accuracy per camera, and maintaining minimum storage requirements, compared with the most recent reported methods. Experimental results performed on the Weizmann dataset [12] and the IXMAS dataset [13] confirm the excellent properties of the proposed algorithm. For future work, our proposed method can be applied using multi-transform domains, where multi-criteria can be extracted to improve the recognition accuracy.

## 5. References


[1] X. Ji, and H. Liu ”Advances in view-invariant human motion analysis: a review” IEEE Transactions on systems, man, and cybernetics-part c: applications and reviews, vol.40, no.1, pp.13–24, Jan. 2010.

[2] L. Sigal, S. Bhatia, S. Roth, M. Black, and M. Isard, “Tracking loose-limbed people,” IEEE CVPR, Washington, DC, vol. 1, pp. 421–428, 27th Jun. to 2nd Jul. 2004.

[3] F. Caillette, A. Galata, and T. Howard, “Real-time 3-D human body tracking using variable length Markov models,” in Proc. Brit. Mach. Vis. Conf., Oxford, U.K, pp. 469–478, Sept. 2005.

[4] C. Menier, E. Boyer and B. Raffin “3D skeleton-based body pose recovery,” in Proc. 3rd Int. Symposium on 3D Data Process., Visualization, and Transmission, Chapel Hill, NC, pp. 389–396, Jun. 2006.

[5] A. Fossati, M. Dimitrijevic, V. Lepetit and P. Fua “Bridging the gap between detection and tracking for 3D monocular video-based motion capture,” IEEE CVPR, Minneapolis, MN, pp. 1–8, Jun. 2007.

[6] D. Weinland, E. Boyer, and R. Ronfard “Action recognition from arbitrary views using 3D exemplars,” IEEE ICCV, Rio de Janeiro, pp. 1–7, Oct. 2007.

[7] P. Yan, S. M. Khan, and M. Shah “Learning 4D action feature models for arbitrary view action recognition”, IEEE CVPR, Anchorage, AK, pp. 1–7, Jun. 2008.

[8] F. Bobick and J. W. Davis, “The recognition of human movement using temporal templates,” IEEE Trans. on PAMI, vol. 23, no. 3, pp. 257–267, 2001.

[9] M.-K. Hu, “Visual pattern recognition by moment invariants,” IEEE Transactions on Information Theory, vol. 8, no. 2, pp. 179–187, 1962.

[10] G. Srivastava, H. Iwaki, J. Park and A. C. Kak "Distributed and lightweight multi-camera human activity classification," 3rd ACM/IEEE Intern. Conf. on Distributed Smart Cameras, Como, Italy, pp.1–8, 30th Aug. to 2nd Sept. 2009.

[11] J. Yang, D. Zhang, A. F. Frangi and J.-Y. Yang “Two-dimensional PCA: A new approach to appearance-based face representation and recognition”, IEEE Tran. on PAMI, vol.26, no.1, pp.131-137, Jan. 2004.

[12] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri "Actions as space-time shapes," IEEE ICCV, Beijing, China, vol. 2, pp.1395–1402, Oct. 2005.

[13] http://4drepository.inrialpes.fr/public/datasets; last retrieved on Sept. 3, 2010.

[14] http://www.wisdom.weizmann.ac.il/~vision/spacetimeactions.html, last retrieved on Sept. 3, 2010.

[15] S. Ali, and M. Shah, "Human action recognition in videos using kinematic features and multiple instance learning," IEEE Trans. on PAMI, vol. 32, no. 2, pp. 288–303, Feb. 2010.

[16] J. C. Niebles and L. Fei-Fei. “A hierarchical model of shape and appearance for human action classification”, IEEE CVPR, Minneapolis, MN, pp. 1–8, Jun. 2007.

[17] C. Yuan, X. Li, W. Hu, and H. Wang ”Human action recognition using pyramid vocabulary tree”, the 9th ACCV, Xi’ an, China, pp. 527–537, Sept. 2009.

[18] W. Yang, Y. Wang, and G. Mori, “Human action recognition from a single clip per action”, 2nd MLVMA (at ICCV), Japan, Sept. 2009.

[19] E. Shechtman, and M. Irani, "Space-time behavior based correlation," IEEE CVPR, San Diego, California, vol.1, pp. 405–412, Jun. 2005.

[20] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action MACH a spatio-temporal maximum average correlation height filter for action recognition," IEEE CVPR, Anchorage, AK, pp.1–8, Jun. 2008.

[21] J. Liu, and M. Shah; "Learning human actions via information maximization" IEEE CVPR, Anchorage, AK, USA, pp.1–8, Jun. 2008.

[22] K. K. Reddy, J. Liu, and M. Shah "Incremental action recognition using feature-tree," IEEE ICCV, Kyoto, Japan, pp.1010–1017, 29th Sept. to 2nd Oct. 2009.

[23] D. Weinland, R. Ronfard, and E. Boyer, “Free viewpoint action recognition using motion history volumes,” Computer Vision and Image Understanding, vol. 104, pp. 249–257, Nov. /Dec. 2006.

[24] F. Lv and R. Nevatia, “Single view human action recognition using key pose matching and viterbi path searching,” IEEE CVPR, Minneapolis, Minnesota, USA, pp. 1–8, Jun. 2007.