RECOGNIZING HUMAN ACTIONS BASED ON MOTION INFORMATION AND SVM

Hongying Meng Nick Pears Chris Bailey

Department of Computer Science, The University of York, UK

ABSTRACT

In this paper, we propose a new system for human action recognition with a view to applications in security systems, man-machine communications and intelligent environments. Our system is based on very simple features in order to achieve high-speed recognition in real-world applications. We have chosen three main techniques to build a system that can work in real-time. Firstly, we choose Motion History Images and related features. Secondly, we use a template matching methods instead of state-space methods that need expensive modelling processes; finally, we use linear classifier support vector machine (SVM) for fast classification. Experimental results show that this system can achieve good performance in human action recognition in real-time embedded applications, such as intelligent environments.

1. INTRODUCTION

Ambient Intelligence (AmI) reflects an emerging and popular field of research and development that is oriented towards the goal of "intelligent" or "smart" environments that react in an attentive, adaptive, and active way to the presence and activities of humans and objects in order to provide intelligent/smart services to the inhabitants of these environments.

An environment is said to be "perceptive" when it is capable of recognizing and describing things, people and activities within its volume. Input can be obtained from sensors for sound, images, and haptics. Video camera or mobile video is easily obtained and can be used for monitoring human events.

Recognizing actions of human actors from image sequences is also an important topic in computer vision with many fundamental applications in video surveillance, video indexing and social sciences. Event detection in video is becoming an increasingly important application for computer vision, particular in the context of activity recognition (1).

Previous work on motion descriptors uses positions and velocities of human body parts (7), but such information is difficult to extract automatically during unrestricted human activities.

For human activity or behaviour recognition, most efforts have been concentrated on using state-space method (5) to understand the human motion sequences (2,4,6,9,13). However, these methods usually need intrinsic nonlinear models and do not have a closed-form solution. As we know, nonlinear modelling also requires searching for a global optimum in the training process, which requires complex computing iterations.

In this paper, we propose a new human action recognition system that is both fast and accurate. It is designed for applications in a security system, man-machine communication, and other cases of Ambient Intelligence. The rest of this paper is organised as follows: In section 2, we will give an introduction to some related work. In section 3, we give a brief overview of our system. In section 4, the detailed techniques of this system are explained including motion features and SVM classifier. In section 5, some experimental results are presented and compared. Finally, we present some discussion and the conclusions.

2. RELATED WORKS

Aggarwal and Cai (1) present an excellent overview of human motion analysis. Of the appearance based methods, many approaches are domain specific or have constraints on the environment as well as the type of motion that can be detected (10).

Recently, template matching has gained more and more attentions. Bobick and Davis (2) use motion-energy images (MEI) and motion-history images (MHI) to recognize many types of aerobics exercises. While their method is efficient, their work assume that the actor is well segmented from the background and centred for the detector.

Schuldt et al (12), proposed a method for recognizing complex motion patterns based on local space-time features in video and demonstrate such features can get good classification performance. They construct video representations in terms of local space-time features and integrate such representations with SVM classification schemes for recognition. The presented results of action recognition justify the proposed method and demonstrate its advantage compared to other relative approaches for action recognition.

Ke et al (8), studies the use of volumetric features as an alternative to the local descriptor approaches for event detection in video sequences. They generalize the notion of 2D box features to 3D spatio-temporal volumetric features. They construct a real-time event detector for each action of interest by learning a cascade of filters based on volumetric features that efficiently scans video sequences in space and time. This event detector recognizes actions that are traditionally problematic for interest point methods such as smooth motions where insufficient space-time interest points are available. Their experiments demonstrate that the technique accurately detects actions on real-world sequences and is robust to changes in viewpoint, scale and action speed.

Weinland et al (14) introduces Motion History Volumes (MHV) as a free-viewpoint representation for human actions in the case of multiple calibrated, and background-subtracted, video cameras. They present algorithms for computing, aligning and comparing MHVs of different actions performed by different people in a variety of viewpoints. Their results indicate that this representation can be used to learn and recognize basic human action classes, independently of gender, body size and viewpoint.

We note that the feature vector in these two methods is very expensive to be constructed and the learning process is difficult to do because it needs a big data set for training.

More recently, Wong and Cipolla (15, 16) proposed a new method to recognise primitive movements based on the motion gradient orientation image directly from images. This process extracts the descriptive motion feature without depending on any tracking algorithms. So it means computational overheads due to tracking can be reduced and the assumptions tracking algorithms usually make can be relaxed. By using a sparse Bayesian classifier, they obtained good classification results for human gesture recognition.

Ogata et al (11) proposed another efficient technique for human motion recognition based on motion history images and an eigenspace technique. In the proposed technique, we use two feature images and the eigenspace technique to realize high-speed recognition. The experiment was performed on recognizing six human motions and the results showed satisfactory performance of the technique. Note that the eigenspace still needs to be constructed and sometimes this is difficult.

In this paper, we propose a new system for human action recognition. This system will be applied in security systems, man-machine communication, and other cases in Ambient Intelligence. Our system is based on simple features in order to achieve high-speed recognition in real-world applications.

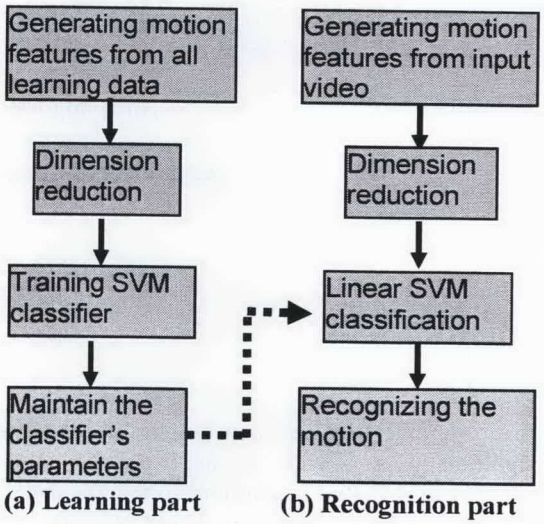

Figure 1: SVM based recognition system

3. OVERVIEW OF OUR RECOGNITION SYSTEM

A suitable classifier will be the core of our recognition system. In this paper, we propose a fast human action recognition system based on a SVM. There are three main reasons for us to choose SVM. Firstly, SVM is a linear classifier, which means it will be very easy and simple in the classification part although the learning part is not simple. Secondly, SVM is a classifier that has achieved very good performance in lots of real-world classification problems. Finally, SVM can deal with very high dimensional feature vectors, which means that we can choose the feature vectors without restrictive dimension limits.

A normal recognition system includes two parts: learning part and classification part. These two parts of our recognition system are showed separately in figure 1.

The feature vectors are to be obtained using motion information directly from the input video. It is expected that the feature extraction algorithms and dimension reduction algorithms should be as simple as possible. The high dimensional feature vector can also be dealt with easily by SVM.

Dimension reduction is an optional part in this system. We prefer to deal with some simple algorithms such as down-sampling or average operations in the feature vector that give very little performance reduction.

The learning part would be processed using video data collected off-line. After that, the obtained parameters for the classifier can be used in a small, embedded computing device such as FPGA or DSP based system, which can be embedded in the application and give real-time performance.

4. TECHNICAL DETAILS

4.1 Motion Features

The recording of human action usually needs huge data space to be stored and it is time consuming to browse the whole video to find the information. It is also difficult to deal with this huge data in detection and recognition. Therefore, several motion features have been proposed to compact the whole motion sequence into one image to represent the motion. The most popular ones are Motion History Image (MHI), Modified Motion History Image (MMHI) and Motion Gradient orientation (MGO).

4.1.1 MHI. A motion history image (MHI) is a kind of temporal template. It is the weighted sum of past successive images and the weights decay as time lapses. Therefore, a MHI contains past images in itself, and the latest image is brighter than past ones.

Normally, a MHI at time k and location (u,v) is defined by the following equation (2):

$$H_\tau(u,v,k) = \begin{cases} \tau & \text{if } D(u,v,k)=1 \\ \max(0, H_\tau(u,v,k-1)-1) & \text{otherwise} \end{cases} \quad (1)$$

where $D(u,v,k)$ is a binary image obtained from subtraction of frames, and τ is the maximum duration a motion is stored. MHI is a multiple values with same size as the frame while Motion Energy Images (MEI) is its binary version. It can easily be computed by thresholding $H_\tau > 0$.

4.1.2 MMHI. Ogata et. al (11) use a multi-valued differential image to extract information about human posture because differential images encode human posture information more than a binary image such as a silhouette image. They proposed a Modified Motion History Image (MMHI) defined as:

$$H_\delta(u,v,k) = \max(f_1(u,v,k), \delta H_\delta(u,v,k-1)) \quad (2)$$

where $f_1(u,v,k)$ is an input image (a multi-valued differential image), H_δ is the modified MHI, and parameter δ is a vanishing rate which is set at $0 < \delta \leq 1$. When $\delta = 1$, it was called as superposed motion image (SMI) which is the maximum value image generated from summing past successive images with an equal weight.

4.1.3 MGO. Motion Gradient orientation (MGO) was proposed by Bradski and Davis (3) to explicitly encode changes in an image introduced by motion events. The MGO is computed from a MHI and a MEI. While a MHI encodes how the motion occurred, a MEI encodes where the motion occurred, the MGO therefore is a concatenation representation of motion (where and how it occurred). MGO can be defined as (3):

$$\phi(u,v) = \arctan\frac{F_v(u,v)}{F_u(u,v)} \quad (3)$$

where $F_u(u,v)$ and $F_v(u,v)$ are the spatial derivatives along u and v direction of the MHI.

All in all, these three features have a common property. They can be generated frame by frame and only three frames need to be stored at any one time in our implementation. The final output associated with each action is an image, which has the same size as the original input frames. This output stores the motion information, which happened during the action process.

Figure 2 is an example to illustrate this. From the original video, we can extract MHI, MMHI and MGO. From these features, although we do not see the whole clip of the action, we can still determine that the girl extended her left arm.

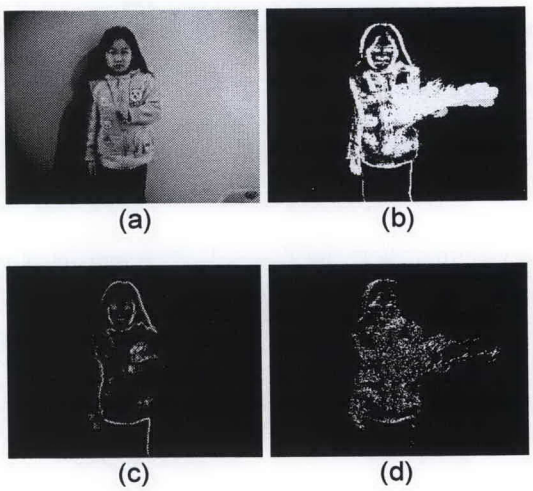

Figure 2: Motion features extracted from human action frames. (a) Original video (b) MHI (c) MMHI (d) MGO

4.2 Dimension Reduction

From the previous section, we get a feature vector from each video of human action. Here, the feature vector is actually a single image. The dimension of these features is still very large. For example, if the size of the video frame is 160×120, the dimension of the features is 19200. Obviously, it is highly redundant feature vector.

Lots of dimension reduction methods can be used here. The most popular one is Principal Component Analysis

(PCA) an this has been used in (11,15,16). Of course, Gabor filter, wavelet transform or other methods can also be used here.

In our system, we choose to use down-sampling methods on the features. In comparison with the above methods, this method needs not only less computation but also less memory and that is very important in embedded hardware based real-time systems. For example, if the dimension of the feature image is 160×120, we can perform a very simple dimension reduction by average all the pixels in a block of 4×4. So the dimension of the feature vectors is reduced to 40×30.

In any case, dimension reduction is an optional part of the system because the SVM can deal with very high dimensional feature vector in the classification.

4.3. Support Vector Machine

Support Vector Machines are a state-of-the-art classification technique with large application fields in text classification, face recognition, genomic classification, etc., where patterns can be described by a finite set of characteristic features.

We use SVM for the classification in our system. This is due to SVM being an outstanding classifier that has shown very good performance on many real-world classification problems. Using arbitrary positive definite kernels provides a possibility to extend SVM capability to handle high or even infinite dimensional feature space.

If the feature vectors are denoted as $\vec{x}$ and its binary labels are denoted as y_i, the norm-2 soft-margin SVM can be represented as a constrained optimisation problem:

$$\min_{\vec{w},b,\vec{\xi}} \quad \tfrac{1}{2}\|\vec{w}\|^2 + C\sum_i \xi_i \qquad (4)$$

s.t.

$$\langle \vec{x}_i, \vec{w}\rangle + b \geq 1-\xi_i, y_i = 1,$$
$$\langle \vec{x}_i, \vec{w}\rangle + b \leq -1+\xi_i, y_i = -1$$
$$\xi_i \geq 0$$

where C is a penalty parameter and ξ_i are slack variables. It can be converted by applying Langrange multipliers into its Wolfe dual problem:

$$\max_{\alpha_i} \quad L_D = \sum_i \alpha_i - \tfrac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle \vec{x}_i, \vec{x}_j\rangle \qquad (5)$$

s.t.

$$0 \leq \alpha_i \leq C$$
$$\sum_i \alpha_i y_i = 0$$

The primal optimum solution for $\vec{w}$ can be represented as a linear combination of the support vectors for which $\alpha_i > 0$,

$$\vec{w} = \sum_i \alpha_i y_i \vec{x}_i \qquad (6)$$

The dual of the optimisation problem can be solved by quadratic programming methods. The final hypothesis is:

$$h_{\vec{w},b}(\vec{x}) = sign(\eta(\vec{x})) = sign(\langle \vec{w}, \vec{x}\rangle + b) \qquad (7)$$

where $\eta(\vec{x})$ is called the confidence coefficient. It should be mentioned here that for classification problems, it is very easy to get a result based on this final hypothesis.

Multiclass SVMs are usually implemented by combining several two-class SVMs. In each binary SVM, only one class is labelled as "1" and the others labelled as "-1". The one-versus-all method uses a winner-takes-all strategy. If there are *M* classes, SVM will construct *M* binary classifiers by learning. During the testing process, each classifier will get a confidence coefficient and the class with maximum confidence coefficient will be assigned to this sample.

$$\tilde{h}(\vec{x}) = i \quad if \quad \eta_i(\vec{x}) = \max_{j=1}^{M} \eta_j(\vec{x}) \qquad (8)$$

In our system, SVM was trained based on features obtained from human action video clips in a training dataset. Generally, we can have several types of actions in a video dataset. Figure 3 shows some examples from a dataset with six types of human actions. Figure 4 shows their features obtained by the above motion extraction algorithms.

These video clips have their own labels such as "walking", "running" and so on. In classification, we actually get a six-class classification problem. At first, we create six binary SVM classifiers, and each of them is related to one of the six classes. For example, there is one SVM classifier related to the class "walking". In the training dataset, the video with label "walking" will have a label "1" in SVM classifier while others have a label "-1" in SVM. Secondly, we will train these SVM classifiers on the learning dataset. The SVM training can be implemented using programs freely available on the web, such as SVM_light (http://svmlight.joachims.org/) (17). Finally, we obtained several SVM classifiers with associated parameters.

In the classification process, feature vectors will be extracted from the input human action video sample. Then all the SVM classifiers obtained from the training process will classify it. Finally, equation 8 will be used to decide which label it should have.

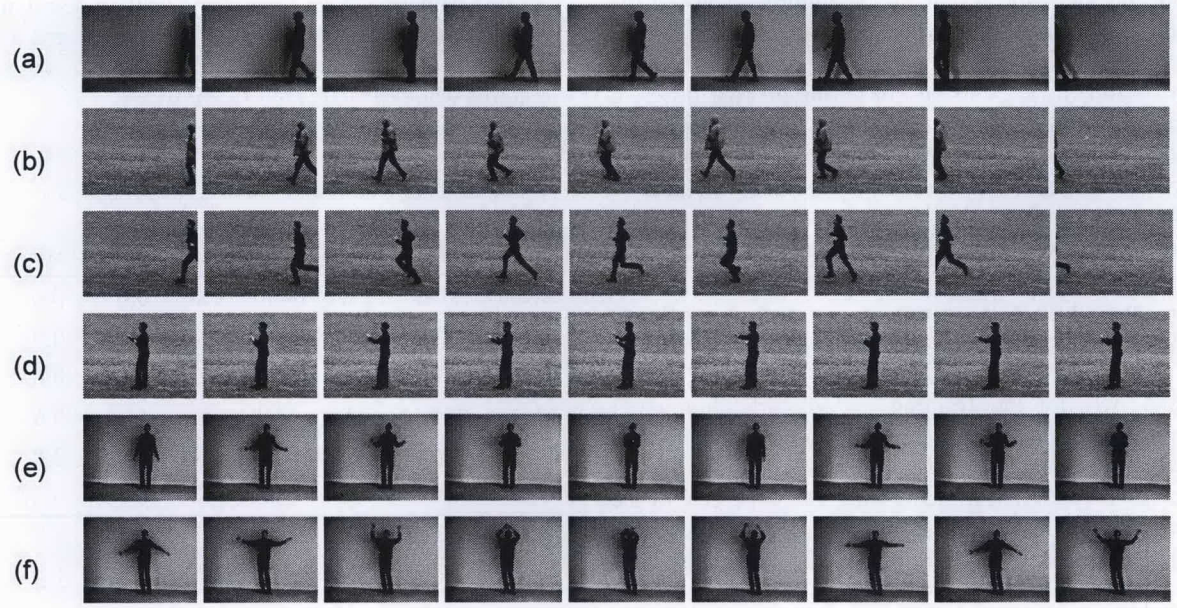

Figure 3: Six types of human action in the database: (a) walking (b) jogging (c) running (d) boxing (e) handclapping (f) hand-waving.

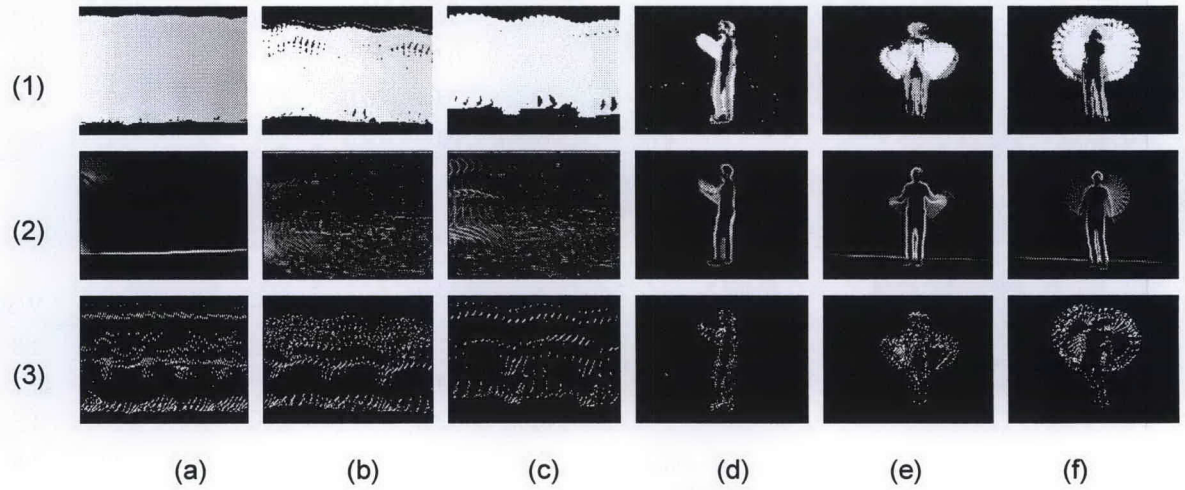

Figure 4: The (1) MHI, (2) MMHI and (3) MGO for the six actions in the dataset: (a) walking (b) jogging (c) running (d) boxing (e) handclapping (f) hand-waving

5. EXPERIMENTAL RESULTS

For the evaluation, we use a challenging human action recognition database recorded by Christian Schuldt (12). It contains six types of human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed several times by 25 subjects in four different scenarios: outdoors (s1), outdoors with scale variation (s2), outdoors with different clothes (s3) and indoors (s4). (See figure 3 for an examples in each type of human action.

This database contains 2391 sequences. All sequences were taken over homogeneous backgrounds with a static camera with 25fps frame rate. The sequences were downsampled to the spatial resolution of 160 × 120 pixels and have a length of four seconds on average. To the best of our knowledge, this is the largest video database with sequences of human actions taken over different scenarios. All sequences were divided with respect to the subjects into a training set (8 persons), a validation set (8 persons) and a test set (9 persons). In our experiments, the classifiers were trained on a training set while recognition results were obtained on the test set.

In our method, we don't need a validation process. But in order to compare our results with others, we test our

method on the same test data set. Figure 4 showed the motion features obtained for the samples in figure 3.

Our experiments are carried out on the all four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes and indoors. In the same manner as paper (8), each sequence is treated individually during the training and classification process. In all the following experiments, the parameters were chosen to be same. The threshold in differential frame computing was chosen as 25 and $\delta = 0.95$ in MMHI.

At first, we did the training and classification on 4 different subsets of the data set. The results can be seen in figure 5. It is the correctly classified percentage on these data subsets that indicates how many percent of the action clips in the testing set were correctly recognized by the system. for all the experiments. It is clear that MHI feature did best work in all the four subset while MGO can not obtain good results for all the four subset. MMHI performance is poorer than MHI.

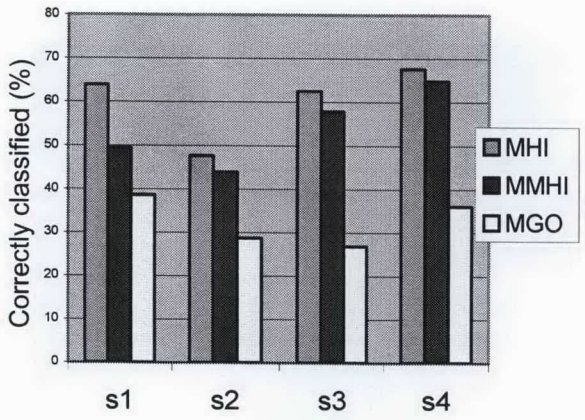

Figure 5: Correctly classified percentage for separate data subset: s1 (outdoors), s2 (outdoors with scale variation), s3 (outdoors with different clothes) and s4 (indoors).

Tables 1 shows the classification confusion matrix based on the method proposed in paper (8) and table 2, 3 and 4 shows the confusion matrix obtained by our method based on different features MHI, MMHI and MGO. The confusion matrixes show the motion label (vertical) versus the classification results (horizontal). Each cell *(i,j)* in the table shows the percentage of class i action being recognised as class *j*. Then the trace of the matrices show the percentage sum of the correctly recognised action while the remaining cells show the percentage of misclassification.

From these tables, we can see that some actions such as boxing, hand clapping and handwaving are easy to recognise, while walking, jogging and running are difficult. The reason is that the later ones are very similar each other both from video sequences or the feature image. Although MGO did very well in hand motion recognition (15), it is not good here. MHI got a better performance than Ke's method based on volumetric features while MMHI is a little bit worse.

TABLE 1. Ke's confusion matrix, trace=377.8

	Walk	Jog	Run	Box	Clap	Wave
Walk	80.6	11.1	8.3	0.0	0.0	0.0
Jog	30.6	36.1	33.3	0.0	0.0	0.0
Run	2.8	25.0	44.4	0.0	27.8	0.0
Box	0.0	2.8	11.1	69.4	11.1	5.6
Clap	0.0	0.0	5.6	36.1	55.6	2.8
Wave	0.0	5.6	0.0	2.8	0.0	91.7

TABLE 2. MHI's confusion matrix, trace=381.2

	Walk	Jog	Run	Box	Clap	Wave
Walk	53.5	27.1	16.7	0.0	0.0	2.8
Jog	46.5	34.7	16.7	0.7	0.0	1.4
Run	34.7	28.5	36.1	0.0	0.0	0.7
Box	0.0	0.0	0.0	88.8	2.8	8.4
Clap	0.0	0.0	0.0	7.6	87.5	4.9
Wave	0.0	0.0	0.0	8.3	11.1	80.6

TABLE 3. MMHI's confusion matrix, trace=369.4

	Walk	Jog	Run	Box	Clap	Wave
Walk	72.9	6.3	15.3	2.1	0.7	2.8
Jog	38.2	19.4	38.2	3.5	0.0	0.7
Run	32.6	12.5	48.6	2.8	0.0	3.5
Box	0.0	0.0	0.0	95.1	2.1	2.8
Clap	0.0	0.0	0.0	25.7	64.6	9.7
Wave	0.0	0.0	2.1	19.4	9.7	68.8

TABLE 4. MGO's confusion matrix, trace=215.1

	Walk	Jog	Run	Box	Clap	Wave
Walk	11.1	27.8	6.9	18.1	16.7	19.4
Jog	9.0	19.4	19.4	12.5	30.6	9.0
Run	2.8	9.7	23.6	26.4	26.4	11.1
Box	0.0	0.0	0.0	69.9	14.7	15.4
Clap	0.0	0.0	0.0	34.0	55.6	10.4
Wave	0.0	0.0	0.0	17.4	47.2	35.4

The above results are obtained based on the feature vector without dimensional reduction. The dimension is 160×120. We perform a very simple dimension reduction by average all the pixels in a block of 4×4.

So the dimension of the feature vectors is reduced to 40×30.

Table 5 and table 6 are the experiment results based on the dimension-reduced feature vectors for MHI and MMHI. It can be found that the performance on MHI is a little bit lower, but the performance for MMHI is a little bit higher.

TABLE 5. MHI's confusion matrix, trace=379.1

	Walk	Jog	Run	Box	Clap	Wave
Walk	59.0	16.0	22.2	0.0	0.0	2.8
Jog	45.8	25.0	27.1	1.4	0.0	0.7
Run	37.5	25.7	36.1	0.0	0.0	0.7
Box	0.0	0.0	0.0	88.8	2.8	8.4
Clap	0.0	0.0	0.0	3.5	91.7	4.9
Wave	0.0	0.0	0.0	10.4	11.1	78.5

TABLE 6. MMHI's confusion matrix, trace=373.6

	Walk	Jog	Run	Box	Clap	Wave
Walk	71.5	8.3	13.2	4.2	0.0	2.8
Jog	36.1	29.9	27.8	5.6	0.0	0.7
Run	27.8	24.3	41.7	2.8	0.0	3.5
Box	0.0	0.0	0.0	95.1	1.4	3.5
Clap	0.0	0.0	0.0	27.1	64.6	8.3
Wave	0.0	0.7	1.4	20.1	6.9	70.8

6. CONCLUSION AND DISCUSSIONS

In this paper, we proposed a new system for fast human action recognition. Potential applications include security systems, man-machine communication, and other cases of Ambient Intelligence. The proposed method does not rely on accurate tracking as most other works do, since most of the tracking algorithms may incur an extra computational cost for the system. Our system is based on simple features in order to achieve high-speed recognition in real-world applications.

From the experiments, it can be seen that this system can give good results. MHI looks better than MMHI in the experiments. The disadvantage for MMHI is that it can only work well in the case of an uncluttered and static background. If there is background motion or noise, it will be recorded in the feature vector that will reduce the performance of the classifications. From the experiments, we can find that dimension reduction gives improved performance for MMHI but does not for MHI. In comparison with Ke's method, we use simple MHI, MMHI or MGO rather than volumetric features in which the dimension of feature vector might be a billion and the performance is little bit better.

In comparison with local SVM methods by Schuldt (12), our feature vector is much easier to obtain because we don't need to find every interest points in each frame. We also don't need a validation dataset for the parameter setting.

For future work, we believe this system can be improved further and be applied in real-world applications. A FPGA based real-time video system will be set up and the algorithms will be modified and optimised based on the hardware limitations such as memory, speed and storage space.

The authors would like to thank DTI and Broadcom Ltd. for the financial support for this research.

REFERENCES

1. Aggarwal J. and Cai Q. 1999.Comput. Visi. and Image Underst., 73(3), 428-440
2. Bobick A. and Davis J. 2001. IEEE Trans. PAMI 23(3), 257-266
3. Bradski G. and Davis J. 2002. Mach. Visi. and App. 13(3) , 174-184
4. Campbell L. and Bobick A. 1995 Proc. of fifth ICCV 624-630
5. Farmer J., Casdagli M., Eubank S. and Gibson J. 1991. Physics D, 51 52-98
6. Gao J., Hauptmann A., Bharucha A and Wactlar H. 2004. ICPR-2004 915 - 918
7. Green R. and Guan L. 2004. Trans. on CAS for Video Technology 14, 179-190
8. Ke Y. and Sukthankar R. and Hebert M. 2005. ICCV2005, 166- 173
9. Nascimento J., M Figueiredo. A. and Marques. J. 2005. BMVC 2005 , 79-86
10. Neumann J., Fermüller C. Aloimonos Y. 2000. IFIP TC5/WG5.10 DEFORM'2000 Workshop, Geneva, Switzerland, 1-11
11. Ogata T., Tan J. and Ishikawa S., 2006. IEICE Trans. Inf. & Syst. 89(1). 281-289
12. Schuldt C., Laptev I. and Caputo B., 2004, ICPR 2004, 32-36
13. Starner T. and Pentland A. 1995 Intl. Work. Auto. Face Gest. Recog. (IWAFGR) '95 189-194,
14. Weinland D, Ronfard R and Boyer E. 2005. ICCV PHI'05
15. Wong S. and Cipolla R., 2005 BMVC 2005
16. Wong S. and Cipolla R. 2005. ICCV-HCI 2005 170-179
17. Joachims T., 1999 "Making large-Scale SVM Learning Practical". Advances in Kernel Methods - Support Vector Learning, Schölkopf B. and Burges C. and Smola A. (ed.), MIT-Press