HUMAN ACTION RECOGNITION BASED ON ENHANCED DYNAMIC TIME WARPING

Bharathram C[1], Dr. S. Chitrakala[2]

[1] Department of Computer Science and Engineering, CEG, Chennai (India)

[2]Associate Professor, Department of Computer Science and Engineering, CEG, Chennai (India)

ABSTRACT

Applications of Human Action recognition (HAR) has started to foray into Gaming, Human computer interaction and surveillance. These new applications require newer and quicker methods that resolves user actions at near real-time speeds. This paper proposes an Enhanced Dynamic Time Warping (EDTW) based approach that recognize human action in a given video in two phases. For this, a kinematic model of the actor is constructed and key poses are found using a suitable clustering method. When a new video dataset arrives, key poses are extracted from it. In first phase of EDTW the similarity of poses from new input are compared with other actions in the repository. In the second phase the measures are processed for confidence and the final classification is done, giving an action label as output.

Keywords: Enhanced DTW, Classification, Human Action Recognition, Human Computer Interaction, Video Processing.

1. INTRODUCTION

Human action recognition (HAR), plays one of the most important roles in computer vision. Describing the actions in a video have a large number of applications, and as the amount of video data grows, and our technology becomes more capable, the demand for action recognition grows. Historically, action recognition has mainly been a human task, but automating the process, or parts of it, could provide a number of advantages. Apart from computer vision, Human action recognition has also forayed in to the field of Human Computer Interaction (HCI).

Using marker-based and marker-less approaches to detect track and recognize the humans present in the video. Historically, human action recognition started from marker based system that required the user to wear specialized suite that had dedicated sensors at designated points on the human body. Apart from wearing this suite the user had to be kept in a controlled environment that had predefined background and less or no background clutter.Marker based approaches were deemed unfit for real time usage. With advent of parallel processing and cost effective hardware for acquiring video, real time, marker-less methods for human action recognition came into significant usage in the recent years. Among Marker-less human action recognition came two models for recognizing actions.

They are conditional models and discriminative models. Conditional models used offered higher accuracy in recognizing the human actions with a trade off in computational time and required a reasonably large amount of memory. Discriminative models on the other hand required less time and computational resources with reasonable compromise in the recognition accuracy. But this soon changed with ability to incorporate Depth information of objects and humans in the videos using specialized hard ware such as Microsoft Kinect.

II RELATED WORKS

Di Wu et all [1] proposed a shape based system that extracts user information in form of silhouette. Their system extracts frames from the given video dataset and constructs a rough silhouette from each frames, extracts information from the silhouettes and recognizes the action conveyed by them using corellogram based estimation algorithm. The advantage of this system is that it uses principle component analysis to reduce dimensions and it is computationally less expensive than analytical approaches. But this falls short in the following areas prone to be affected by occlusion and does not capture spatial variations.

Li Liu et all [2] proposed that uses a ‘kinematic based approach by constructing a model by extracting features from a sequence of frames and estimating Human Pose from them. This approach uses Weighted Local Naïve Bayes Nearest Neighbor Classifier for recognizing the final action. Since not all poses in a sequence are discriminative and representative, AdaBoost algorithm was used to learn a subset of discriminative poses. The advantages of this model is that it uses model to effectively represent human poses and Takes only the key poses that convey information, and discards the rest. Limitations of this approach are that it is computationally expensive than Silhouette and Learning Time overhead in case of using Adaboost.

Alexandros Andre Chaaraouiet all [3] presented a system that uses an adaptive approach that works well with ‘Shape based’ and ‘kinematic model based methods of estimating Human Pose from videos. The main advantage of this method was that it has an evolutionary algorithm that gives room for learning new poses. Though it eliminated the need for retaining the system it suffered from limitations such as time taken for preprocessing frames for adaptation, May require up to 50 iterations to converge and Results may change according to random initialization in K-means.

Wei Shen et all [4] presented a system that Uses Fourier temporal Pyramid representation to estimate Human Pose from videos. Uses a new method of learning called ‘Actionlet Ensemble. As Human actions usually involve human-object interactions, highly articulated motions, high intra-class variations, and complicated temporal structures. The final action recognition was done using Hidden Markov model or Neutral networks. The main selling point of this method was that it was robust to temporal misalignment and had high tolerance for noise. Limitations of this algorithm was that works well only on simple activities such as drinking, running etc and it needs some supervision in case of complex tasks.

Jamie Shottonet all [5] proposed a system that allowed us estimate simple depth pixel comparison features and parallelizable decision forests to detect poses in real-time. It uses a Parallel decision forest algorithm to estimate the action of human present in a video dataset. This method preforms well in real time speeds when compared to other methods. To achieve such a performance it requires depth information to be present in the video and often results in lower recognition rate.

Al Mansur et all [6] proposed a system for HAR using a physics-based model that articulates and actuates muscles and consists of joints with variable stiffness. The main advantage is that these features are more discriminative than kinematic features, resulting in a low-dimensional representation for human actions, which preserves much of the information of the original high-dimensional pose. However any abrupt changes in positions cannot be handled with this method.

RavitejaVemulapalliet all [7] have made an approach to HAR using a Kinematic model with a combination of Support vector machine to arrive at a target action class. They used a new skeletal representation that explicitly models the 3D geometric relationships between various body parts using translations and rotations in 3D space. Since 3D rigid body motions are members of the special Euclidean group SE(3), the proposed skeletal representation lies in the Lie group. With the proposed representation human actions can be modeled as curves in this Lie group. The main advantage is that this performed on better than the state of the art skeletal based systems with a higher recognition rate, but required time to converge.

III SYSTEM ARCHITECTURE

3.1Proposed System

This Paper presents an approach that uses combination of kinematic modelling and EDTW. Kinematic modelling is mainly used to represent humans present in the given video and then a Bag of key poses model to represent the sequence of key poses. Thus, a repository is created to store the training data of various selected actions. When a new video is given as input, kinematic model of the human is constructed and action recognition is performed using an Enhanced Dynamic Time Warping (EDTW). The classification algorithm here compares the similarity between each key pose in the repository against each key pose in the new video dataset. The overall similarity is computed in terms of two measures, as the overall bag-wise distance and the posewise similarity. Based on the two measures and the proposed algorithm, the system arrives at an action label output.

3.2 Module Design

The system consists of three major module. They are Preprocessing, Transformation and Classification Module.

3.2.1 Video Pre-Processing Module

This module consists of three sub modules, they are Frame Extraction, background subtraction and feature extraction.

Fig – 1 Proposed System

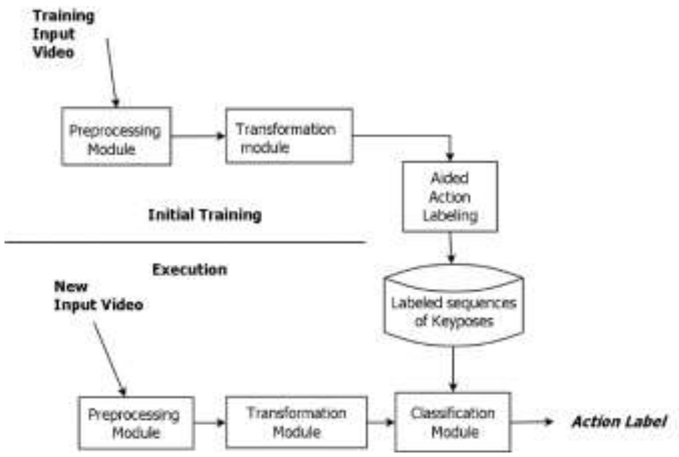


Fig – 2 Frame Extraction

3.2.1.1 Frame Extraction

The frames extracted can be actual frames used in the recording according to the frame rate or further processing toarrive at key frames can be used. If a full length video dataset is used it would be better to use Key Frame extraction. On the other hand if only a video snippet or a small dataset is used normal frame extraction will suffice. The results of the frame extraction is shown in figure 2 which corresponds to a person walking.

3.2.1.2 Background Subtraction

Once the frames are extracted, each frame is taken one by one and the foreground objects are separated from the background clutter. This is done by background subtraction, this removes background from foreground objects of interest, in this case it is the human or actor in that frame.

Say if

Background Image at time $T_0 = B(x,y,t_0)$

Current Image at time $T_n = I(x,y,t)$

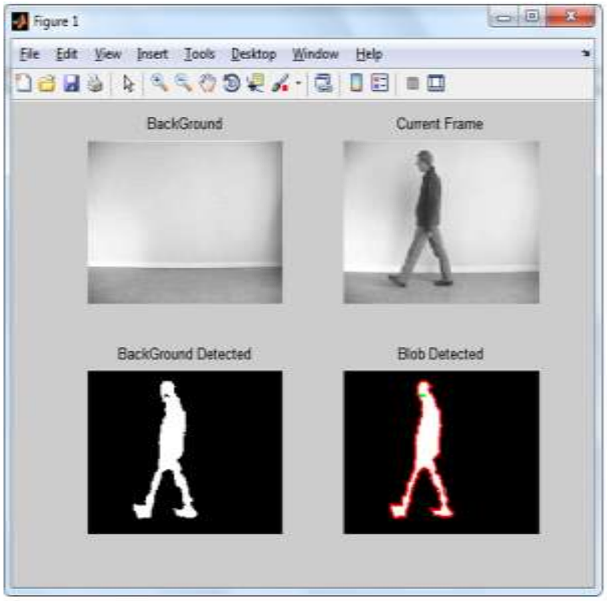
The foreground mask can be obtained by

$$F(T_n) = I(x,y,t) - B(x,y,t_0) \quad (1)$$

An example of background subtraction is shown in the figure 3, the initial frame, the foreground detected, are shown one after another.

3.2.1.3 Feature Extraction

Feature extraction converts the foreground objects present the frames obtained after background subtraction into numerical data to aid in further representation and computation, in this case Gaussian or Fourier pyramidal features is used to extract position information from each frame. The data thus extracted is stored in an interim storage or in a text file. To aid in simplification in representation a csv file can also be used.



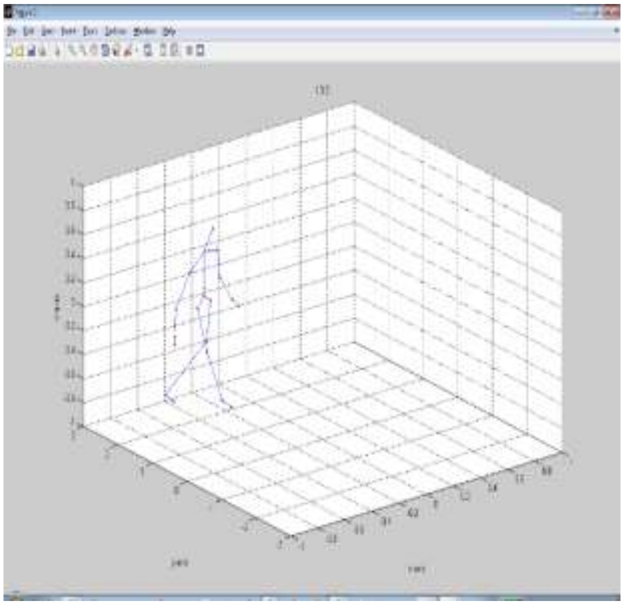


Fig.3. Background Subtraction **Fig.4. Kinematic Model of Actor**

3.2.2 Transformation Module

The transformation module, converts the numerical data into a human model and does the job of dimensionality reduction. After reduction, it converts the models into bag of key poses by transforming the set of numerical data into clusters and storing them in a repository.

3.2.2.1 Kinematic Model Construction

This module is used to convert the numerical data obtained, into human model. Here, the system constructs a kinematic model with 20 points. Each point represents a joint or a part of the actor present in the frame. Thus, each frame will have a 20 point Tuple representing which part moved and to what extent it moved.The model is constructed from the set of feature vectors received from the feature extraction phase, using the depth information as and when available.

3.2.2.2 Clustering

Pose selection is done using a Clustering Technique. Here K-Means or K- Metoids can be used to select poses that convey maximum amount of information and put them in a bag called as bag of key poses. The current systemuses K-means. Now, the system has a set of Human 3D models (X_1, X_2, …,X_n), where each observation is a 20-dimensional vector, k-means clustering aims to partition the n observations into k (= n) sets S = {S_1, S_2, …, S_k} so as to minimize the within cluster sum-of-squares.

The number of centroids varies anywhere from 25 to 50 depending on the temporal intensity of the action. Faster the action being performed, more centroids and slower the action, lesser centroids. The Bags of key poses thus obtained for different actions is collected and Action Labels are assigned. Such labeled bag of key poses are stored in a common repository. This serves as reference for further action recognition task. The Repository in this case are dedicated text files each for an action. The text file, contains all the 20 – point Tuples centroids each with 3 Values (Orientation, X-Co-ordinate, Y – Coordinate). The joint data is stored in text files in .dat format for the ease of further processing.

3.3 Classification Module

The Classification module is responsible for computing the similarity between the centroids in the current bag of poses versus the centroids of all the bags present in the current repository.

3.3.1 Enhanced Dynamic Time Warping

Classification is carried out in two stages, by Enhanced Dynamic Time Warping (DTW), DTW extracts one pose at a time and compares it with all other poses in the repository, calculates the distance between them and stores them in an N x M matrix. Here N is number of key poses in the current action label and M is the number of actions present in the repository.

Once Distance measurement is completed, the system has pose wise similarity between all poses in new input to all poses present in the repository and the overall all distance by which the current bag varies from the bags present in the repository. Now using this distances confidence of the values are measured. The runtime output is presented in figure 8.

Input – Repository of key poses and key poses of current video
Output –Bagwise Distance, Posewise Distance measurements
Algorithm

```
Set N = number of centroids in current dataset
Set M = number of actions in repository
Set Q = number of centroids in repository
Create DIST [N] [M]
Create ODIST[M]
For (n=0; n<N; n++)
{
        Get point (Xn ,Yn)
        For (m=0; m<=M; m++)
        {
        Set DIST [N] [M] = 0
    For(q=0; q<=Q; q++)
```

```
            {
              Get point(Xq , Yq)
              DIST [n] [q] = $\sqrt{(Xn - Xq)^2 - (Xn - Xq)^2}$
              Write Pose-wise similarity
            }
      ODIST[m]= $\frac{\sum_0^N dist(n,m)}{\sum_M m}$
            Write Bag-wise similarity
      }
}
```

3.3.2 Class Confidence Processing

Class confidence measurement, normalizes the distances and helps to identify the sequence of poses which is significantly similar to an action present in the repository. If such a bag of poses is found the action corresponding to that bag can be given as the output label. Here the distance are normalized with respect to their mean, and threshold for confidence is set relative to the variance and the standard deviation of the normalized distribution of the sample space.

Input – List of bag-wise Distances

Output – Action label if a class with high confidence is found.

Algorithm

```
    1.  Get ODIST[M]
    2.  Set Mean= $\frac{\sum_0^M ODIST[m]}{M}$
    3.  Compute Variance of ODIST[M]
    4.  Threshold = 0.10* (Variance )^0.5
    5.  Sort(ODIST[M]) in non-decreasing order
    6.  If((ODIST[1] - ODIST[0]) >= Threshold) then
               Output Label corresponding to ODIST[0].
        Else
    Output("New Action\Actor Encountered");
```

IV EXPERIMENTAL RESULTS

The implementation was done in MATLAB 2014a and in C. First two modules implemented in Matlab script files (*.m). The interim Outputs are written into Images, text files or kept as temporary .mat variables as and when required. The last two modules were implemented in C language to improve the execution speed of numeric intensive parts such as Enhanced Dynamic Time Warping classifier, confidence measurement.

The Dataset used for this system are mainly from Microsoft Action Recognition Datasets [8], Weizmann dataset [9] and UT Kinect [10] Human Action Recognition dataset. The datasets contain videos of 16 actions such as drinking, eating, reading a book, calling, walking, push, pull, throw, writing on a paper, using laptop, etc. All the actions were performed by different actors. Some of these videos were captured using a kinetic device and the stored offline for processing.

4.1 Results

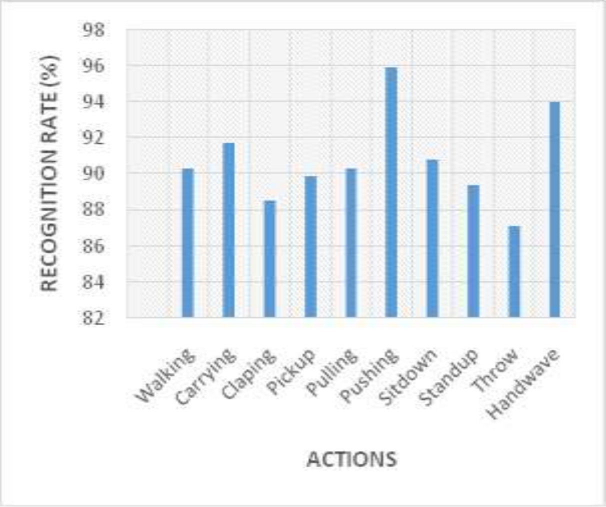
Recognizing actions present in the video narrows down to a classification problem. Instead of traditional measures of precision, the system uses a slightly modified factor called as Recognition rate. This determines the percentage of actions the system classifies correctly out of the total number of input actions with respect to the key poses present in repository.

$$\mathrm{RR} = \frac{\text{No.of instances of correct positive recognition}}{\text{Total no. of positive recognitions}} \tag{3}$$

$$\mathrm{Recall} = \frac{\text{No. of instances of positive recognition found}}{\text{Total no.of relavent Input Instances}} \tag{4}$$

The results for action recognition in terms of recognition rate is summarized in Table 1. Recognition rate is the equivalent to precision in this case. From figure 6, it can be observed that the current system performs on par with the existing system. However, the previous implementation has relied fully on highly sophisticated platform for their execution. Since parts of this implementation runs on relatively simple environments such as C and windows batch scripts, this makes more suitable for running on less powerful machines and can execute at near real time speeds.

The results for action recognition in terms of recall is summarized as follows in Table 1. From the figure 7, it can observed that recall for actions similar to existing action in the system is lower than that of action that are considerably distinct from others. From the results, it can be observed that the system performs on par with the existing systems but does so at near real time speeds on a relatively lighter platform. Also, the system performs well against standard/regular actions at higher rates of accuracy. The system does tend to misclassify some videos which is due to the fact that there are some actions that have common poses between them.



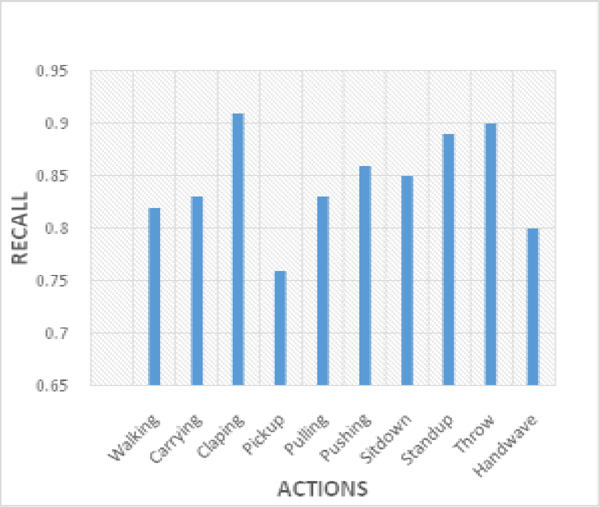


Figure 6 – Recognition Rate for different actions **Figure 7 – Pose Recall Measures obtained**

Table – 1 – Recognition Rate and Recall Measures

Action	**Recognition Rate (%)**	**Recall (%)**
Walking	90.31	81.67
Carrying	91.71	82.66
Clapping	88.57	90.67
Pickup	89.91	75.69
Pulling	90.29	83.4
Pushing	95.91	86.36
Sit down	90.83	84.51
Standup	89.42	88.73
Throw	87.15	89.57
Hand wave	94.04	80.32

V CONCLUSION AND FUTURE WORK

Thus in the current approach, the system is able to successfully recognize the action performed by the humans present in the video using kinematic modeling and Enhanced Dynamic Time Warping and achieves an on-par performance with the current systems. Final Classification results were obtained at near real-time speeds with an on-par accuracy in recognizing the actions. The system can be further extended to recognize complex actions, incorporate learning capabilities to learn new actions, classify actions that involve interaction with objects and other humans. The system can also be improved in terms of being more robust to detect similar actions.

REFERENCES

[1] Alexandros Andre Chaaraoui, Student Member, IEEE, and Francisco Flórez-Revuelta, Senior Member, IEEE,Adaptive Human Action Recognition With an Evolving Bag of Key Poses, in IEEE Transactions On Autonomous Mental Development, Vol. 6, No. 2, June 2014.

[2] Al Mansur, Yasushi Makihara, and Yasushi Yagi, Inverse Dynamics for Action Recognition, in IEEE Transactions On Cybernetics, Vol. 43, No. 4, August 2013

[3] Li Liu, Ling Shao,Senior Member, IEEE, Xiantong Zhen, and XuelongLi,Fellow, IEEE, Learning Discriminative Key Poses for Action Recognition, In IEEE Transactions On Cybernetics, Vol. 43, No. 6, December2013

[4] Di Wu,Student Member, IEEE,and Ling Shao,Senior Member, IEEE, Silhouette Analysis-Based Action Recognition Via Exploiting Human Poses, IEEE Transactions On Circuits And Systems For Video Technology, Vol. 23, No. 2, February 2013.

[5] Jamie Shotton,Senior Member, IEEE, Ross Girshick, et al, Efficient Human Pose Estimationfrom Single Depth Images, in IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 35, No. 12, December 2013.

[6] Zia Moghaddam and Massimo Piccardi, Senior Member, IEEE, Training Initialization of Hidden Markov Models in Human Action Recognition in IEEE Transactions On Automation Science And Engineering, Vol. 11, No. 2, April 2014.

[7] RavitejaVemulapalli, Felipe Arrate, and Rama Chellappa, Human Action Recognition by Representing 3D Human Skeletons as Points in a Lie Group, CVPR, 2014.

[8] Microsoft Action Recognition Datasets, http://research.microsoft.com/enus/um/people/zliu/ActionRecoRsrc/

[9] Weizmann Human Action Recognition Datasets, http://www.wisdom.weizmann.ac.il/~vision/VideoAnalysis/Demos/SpaceTimeActions/

[10] UT Kinect Human Action Recognition, http://cvrc.ece.utexas.edu/KinectDatasets/HOJ3D.htm

Biographical Notes:

Mr. Bharathram C, Is currently pursuing his Master’s degree in Computer Science and Engineering at College of Engineering – Guindy. He has received his Bachelor’s degree (B.E) in engineering from Anna University in electronics and communication engineering at 2006.

Dr. S. Chitrakala is working a associate professor in department of Computer Science of Engineering at College of Engineering – Guindy, Chennai.