Automatic Temporal Location and Classification of Human Actions based on Optical Features

Seyed Ali Etemad
Dept. of Systems and Computer Eng.
Carleton University
Ottawa, Canada

Pierre Payeur
School of Information Tech. and Eng.
University of Ottawa
Ottawa, Canada

Ali Arya
School of Information Tech.
Carleton University
Ottawa, Canada

***Abstract*—This paper presents a method for automatic temporal location and recognition of human actions. The data are obtained from a motion capture system. They are then animated and optical flow vectors are subsequently calculated. The system performs in two phases. The first phase employs nearest neighbor search to locate an action along the temporal axis taking into account both the angle and length of the vectors, while the second classifies the action using artificial neural networks. Principal Component Analysis (PCA) plays a significant role in discarding correlated flow vectors. We perform a statistical analysis in order to achieve an efficient, adaptive and targeted PCA. This will greatly improve the configuration of flow vectors which we have used to train both the locating and classifying systems. Experimental results confirm the significance of our proposed method for locating and classifying a specific action from among a sequential combination of actions.**

Keywords-temporal location; classification; human actions; neural networks; principal component analysis.

I. INTRODUCTION

Analysis and recognition of human motion is an essential element in security, traffic, sports, multimedia, and biomedical technologies. During the past decade an extensive amount of research has been carried out with the goal to create a robust system, capable of recognizing human actions and a variety of different tools and techniques have been employed.

Optical flow provides unique features which has made it very suitable for training systems for human action recognition. In [1-8] the utilized data is based on optical flow vectors or some refined format of flow features. The Lucas-Kanade algorithm for optical flow computation has shown to be most common and effective when optical data are available. Some other types of motion records such as motion capture data [9,10] and data acquired from accelerometers [11] have also been employed.

When using optical flow features, the excessive number of correlated data must be reduced. PCA (Principal Component Analysis) is the most common tool [3-6] for this purpose. Other techniques such as Adaboost [2] and flow histograms [1,6,8] have also been suggested to create stronger features. Each motion frame is usually divided into *n* subsections (channels) and the refining algorithm is performed on each individual channel. The outcome for each channel is then used to train the system. For example, [8] the authors create a flow histogram for three channels representing three vertical slices for each frame. Ikizler et al. [1] divide the frames into 9 equal rectangles and subsequently form histograms for each of the 9 channels. This type of partitioning the frames does not take into account the fact that some channels, for instance the channels corresponding to the corners of the image, are occupied by insignificant or almost no data. In the training process however, all channels are employed with equal weight and influence on the system.

There are basically two problems to tackle when dealing with recognition of human actions. The first is to locate a specific action along the temporal axis. This means, in a sequential combination of actions performed by an actor, the goal is to determine when an action begins. A sliding search window may be an option [1] for this purpose, yet since for each new position of the window, the entire classification process must take place, it shows to be very time consuming. The other main problem is to classify the selected action. Varieties of different tools have been utilized for action classification. Hidden Markov models (HMM) are one of the most common tools [4,5,12,13,14]. Support Vector Machines (SVM) [1,8,15] and K-Nearest Neighbor (K-NN) [3,6,7,16] have also been utilized. The main tool which we have employed for classification of actions, Artificial Neural Networks (ANN), have also been used largely for action recognition [17-21]. Kornprobst et al. in [17] show that visual data used to train neural networks are an effective and efficient means for human action recognition. In [18] Babu et al. employ MHI (Motion History Image) and train neural networks for the recognition task. Self organizing neural networks have also been utilized by Kuniyoshi and Shimozaki [19,20]. Last but not least, in [21] Theodoridis and Huosheng use a variety of different neuron/layer MLP networks along with different training functions to classify human actions and compare the performance for different situations.

The research reported here tries to address the two mentioned problems for recognition of human actions. We have made use of the Lucas-Kanade algorithm for optical flow

978-1-4244-4131-0/09/$25.00 ©2009 IEEE

computation and PCA for data refinement. In this research we have not used the conventional linear partitioning of the images, and have proposed an effective method to tackle this problem by creating non-identical and dynamic channels based on statistical analysis in order to configure efficient, adaptive and targeted channels. The classification system performs in two phases. The first phase employs nearest neighbour search and locates specific actions temporally. For this purpose both the length and angle of the vectors are taken into account. Then the main classification system takes action as the second phase, classifying the located actions into pre-defined classes. Phase two utilizes artificial neural networks for human action recognition purposes.

II. Optical Flow Features, Dynamic Channels, and Targeted PCA

The data used to create both the recognition and classification systems are based on optical flow computation of an animated skeleton of the actor performing the actions. This benefits our system by simplifying our research such that the clothing variations and background is no more an issue to deal with. In most literature on this topic, the background and actor clothing is chosen such that the separation of the actor from the background environment is simplified [4,5,7]. The motion capture data are obtained by a Vicon system based in Carleton University, and the motion frames are extracted via Autodesk Maya 3D animation software. Once the data acquisition is accomplished, the optical flow vectors are calculated using the very practical and popular Lucas-Kanade algorithm. Fig. 1 shows the calculated optical flow vectors for frame 10 of a walking and jumping sequence.

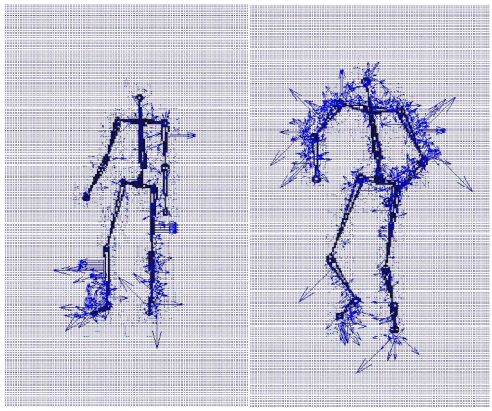

Figure 1. Optical flow vectors for walking sequence (left) and jumping sequence (right)

The number of the obtained vectors are considerably large, thus it must be reduced. Using PCA, enhanced non correlated optical motion features are extracted.

As discussed earlier in section 1, the images are usually divided into similar sections with equal areas, and PCA is performed on each section for flow vectors. When applying this approach, all channels will be equally weighed even for channels which do not contain flow features critical to characterizing the action, for example the channels in the corners of the images. Also in this fashion, the motion of a body part such as the arm might be partially represented by one channel and partially by another. As a result a specific channel which contains some of the arm motion features might also contain the motion features for the spine section. Performing PCA on this channel will then result in vectors representing both the arm and spine. Such situations are likely to happen for other parts of the body as well, which will be confusing for the recognition system. Another possibility is that one of the two important body parts represented by a specific channel be totally ignored since they might contain correlated motion features. This situation will result in loss of features that may be critical for recognition of actions.

Our proposition for tackling the channeling problem is to introduce non-similar dynamic channels for the motion. Prior to the optical flow computation, the center mass of the skeleton is measured and a fixed 150 by 250 pixels box with same center mass is applied, discarding any pixel outside the box. The size of this box is selected based on the lengthiest poses available to guarantee that all actions remain inside the box. Optical flow is computed for the box only, guaranteeing that the skeleton is in the center of the image, thus reducing the number of flow vectors for the skeleton to some extent as shown is Fig. 1. Reduction of flow vectors happens due to the fact that by centralizing the actor for all frames, some body parts such as the spine will seem at rest although they are not, and only critical curl movements will be taken into account as motion. The mask is then configured on each 150 by 250 window based on two concepts: the anatomical shape of the body and statistical analysis of the locations at which most motion features appear in with respect to the overall configuration of the image pixels. For this purpose the flow vectors are calculated within the box for all frames of the sequence. For each pixel the sum of the magnitudes of all vectors associated with that pixel during an action is calculated and normalized based on all pixels such that they vary from 0 to 255. Fig. 2 (left) shows the outcome of this analysis where brighter pixels show more motion. Based on the anatomical shape of the body, each channel is defined with respect the center mass of the body and the borders of the 150 by 250 box. The channels are configured such that they represent a block-style version of human body. Six channels are created representing six important sections of the body: head, left arm, right arm, spine, left leg, and right leg. The size of the channels will change with the change in perspective size of the actor body, with actor height, and with different poses that are created throughout an action. Fig. 2 (middle) illustrates the mask applied on a practical image where the flow vectors have been computed. In Fig. 2 (right), the adaptiveness of the mask for a different pose and different perspective height and width is demonstrated. Body parts may indeed cross the boundaries and depart into non-relative channels. This situation is unavoidable in 2D images, nevertheless we have minimized this issue with our approach. Also this technique would reduce the negative effect of possible variation in test actor height and bulkiness due to its adaptation with the volume of the body.

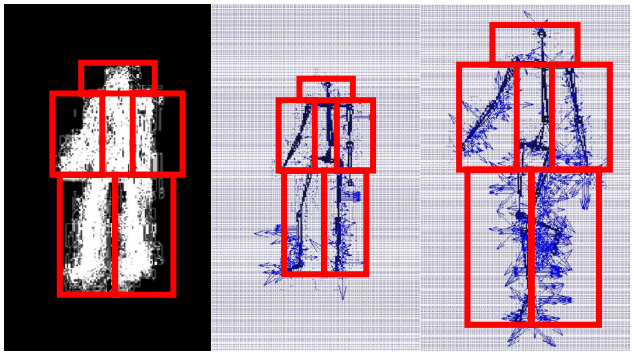

Figure 2. Channeling mask for PCA process – mask formation (left), mask adaptation (middle, right)

Subsequent to applying the channel masks, PCA is performed on each of the 6 channel for all the flow vectors within. The result is a number of vectors weighted with respect to their correlation. The sum of these vectors is a single vector representing that channel. These vectors are later used for locating the actions temporally and for classification.

III. Temporal Location of Actions

The vectors representing each channel in each frame compose the training data for the location and classification subsystems. To create the location subsystem, nearest neighbor search is selected due to its fast routine and accurate capability in recognition of similar features.

For training the nearest neighbor classifier, the temporal location where each action initiates is labeled. The optical flow vector for each channel contain both amplitude and phase denoted by (l_i,θ_i). The Euclidean distance between the test vector and the training vectors are calculated by the polar distance formula (1) where the τ subscript represents the training features and c represents the channel number. j represents the action class (walk, run, or jump) and i denotes the frame number. The distance d is computed for all i and j values twice: once for the starting point of each action class j and once for the ending instance. The i which returns the least value of d for each round of calculations indicates the starting and ending point of the action.

$$d = \sum_{c=1}^{6} \sqrt{l_{\tau,c}{}^2 + l_{i,j,c}{}^2 - 2 l_{\tau,c} l_{i,j,c} \cos\left(\theta_{\tau,c} - \theta_{i,j,c}\right)} \qquad (1)$$

The reason that this procedure is not used to classify the action along with the temporal locating of the action is that this technique simply employs the beginning and ending instance channels, and the frames in between can be of any sort. The goal of this search is to find the frames which appear to be the most similar to the start and end instances of any of the three classes. Also KNN is not adaptive compared to ANN which would be specialized to learn the alteration of flow vectors for a specific action.

IV. Action Classification using Neural Networks

Classification of the actions is carried out by means of ANN. By constructing a single neural network and training it with the three types of action the classification task can take place. This network employs 12 inputs one for each component of each channel, and 1 output, determinant of the action class. The input values are computed by differentiation of consecutive frame values, while the output value is 1, 2, or 3 representing the classes of action which the difference vectors belong to. As it will be demonstrated in section 5, this network will not be very precise since differentiating between the three classes of action which hold many similarities is confusing for the network. Our proposed method is to create a different neural network for each action class. These networks play the role of anticipators. The frame span measured by the KNN is employed by all three networks, where each frame is used as the input and the consecutive frame plays the role of the output. Finally the mean square error (MSE) is calculated and the network producing the least MSE determines the class of action. The classification process is illustrated in Fig. 3.

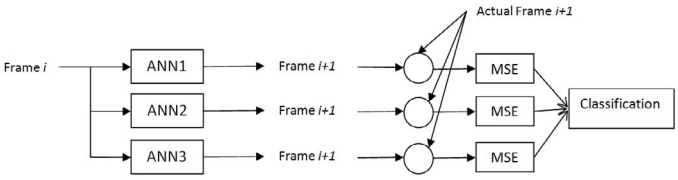

Figure 3. Classification process

For this research, the very practical, yet efficient multilayer perceptron (MLP) with backpropagation learning technique is utilized. The networks hold 12 inputs and 12 outputs, one for each component of each of the 6 channel vectors, and two hidden layers are configured.

V. Experimental Results

The goal of this research is to create an automatic system capable of locating and classifying human actions using different motion sequences. The overall functionality of the system is based on the discussed topics in sections 2, 3, and 4. The flow diagram of the system is presented by Fig. 4 which shows the different steps of this process. Initially optical flow is computed for the entire images and the dynamic channels are then configured based on statistical analysis and human figure. PCA is applied on each channel and the resulting vectors are used to train the nearest neighbor locator as well as three neural networks. Then for every test set, optical flow is computed, dynamic channels are applied, PCA is applied on each channel, and the results are employed by nearest neighbor locator for temporal location. The segmented action is then fed through the anticipators for classification of the located action.

The dynamic channeling process is carried out based on section 2. To evaluate the contribution of this technique compared to fixed size static channels, the images were also divided into six non-adaptive channels where the width of each image is divided into 2 sections and the height is divided

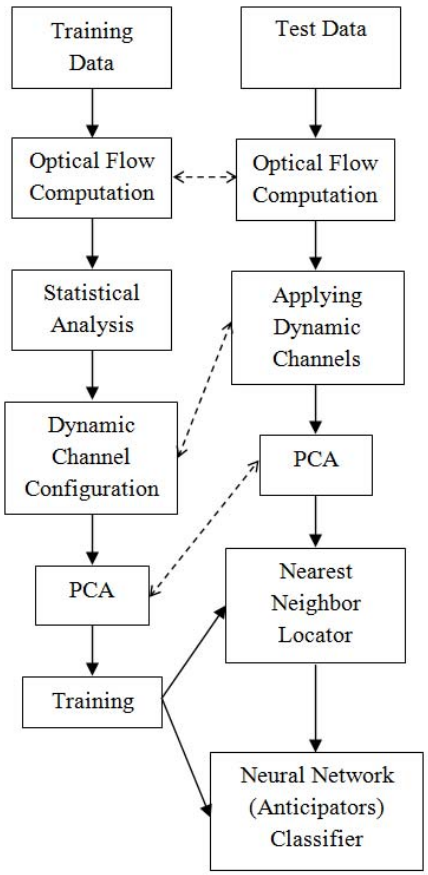

Figure 4. Flow diagram of the system. Dashed lines show similar procedures of the steps.

into 3 sections. Following the channeling procedure, PCA is applied to each channel which results in a single vector representing each channel for both techniques. The nearest neighbor search is then carried out, locating the most likely frames representing the start and end temporal instance of an unknown action. Table 1 shows the results for temporal location of actions for each of the two channeling techniques. Six 100 frame sequences are tested where each action class is included in two of the sequences. The rest of sequence is filled with actions other than the three action classes. The error is calculated by the sum of differences in frame number with respect to the correct temporal instance of the beginning and ending of each action, divided by the total number of frames of each action.

TABLE I. TEMPORAL LOCATION USING FIXED AND DYNAMIC CHANNELS

Action	Error: Fixed Channels	Error: Dynamic Channels
Walk	5.97%	5.97%
Jump	3.99%	3.99%
Run	6.38%	5.07%
Average	5.45%	5.01%

It can be observed from Table 1 that the contribution of dynamic channeling is not very significant to the search for temporal location of the actions as the search for both walk and run remained unchanged. Overall, some improvement is observed while the accuracy of the nearest neighbor search for both techniques is acceptable and precise.

The classification of the located actions is then carried out using neural network for both techniques. Table 2 presents the results where both a single neural network and three neural networks in the form of anticipators have been tested.

TABLE II. CLASSIFICATION OF ACTIONS USING FIXED AND DYNAMIC CHANNELS

Action	Classification Results: Dynamic Channels	Classification Results: Fixed Channels	Classification Results: Dynamic Channels
Neural Network	**Single**	**Triple (Anticipators)**	**Triple (Anticipators)**
Walk1	Run	Walk	Walk
Walk2	Run	Run	Walk
Jump1	Jump	Jump	Jump
Jump2	Jump	Jump	Jump
Run1	Walk	Walk	Walk
Run2	Run	Run	Run
Accuracy	50.00%	66.67%	83.33%

Table 2 clearly demonstrated the effect of employing dynamic channels as well as using neural networks in the form of anticipators. While a single neural network shows an accuracy of 50%, three specialized neural networks one for each class increases the accuracy by more than 16.67%. Also the effect of dynamic channels is significant (16.67% improvement) for classification as opposed to the temporal search where it did not affect the results by a considerable margin. The reasons for these drastic improvements in classification results are the facts that 1) when three neural networks are used, the networks are not confused by the different classes of action used to train them 2) dynamic channels simplify and specialize the flow vectors in each channel compared to the case where static and equally sized channels are utilized. From Table 2 we can conclude that each of these measures improves the classification accuracy by 1/6.

In regards to the runtime of the proposed method, the system is far from real-time. Yet the speed of the system is significantly higher compared to the practical method. The application of dynamic channels omits the need for implementation of PCA for up to 2/3 of the area of the images, resulting in reduction of the lengthy PCA runtime by nearly 2/3 of the original method where PCA is applied on the entire images using fixed size channels. This impact varies based on the perspective of the figure which results in faster computations from between 1.5 to 2 times the original speed. The same impact is visible for the location task where fewer channels result in faster temporal location of actions. The use of 3 separate neural networks in the form of anticipators

instead of a global network does not significantly impact the runtime. Training three different networks with a fixed number of data sets for each, consumes almost the same amount of time as one single network takes to be trained by all the data sets used to train each of the three. This means the training time which is a very lengthy process is almost equal in both cases. The testing of the data using three anticipators, however, takes triple the time required to test the data using one network. Yet since the testing is a very fast procedure, the difference in runtime is negligible. Thus the overall runtime of the system is significantly decreased compared to the practical optical flow/neural network methods.

VI. CONCLUSION

In this paper, we proposed a method for automatic classification of actions. The actions were located from within a sequential combination of actions. The process for temporal location of actions was carried out using nearest neighbor search. The outcome was employed by three neural networks in the form of anticipators, each of which were earlier trained by features of a specific class of action, and the network producing the least mean square error determined the actions class. It was demonstrated that using separate specialized neural networks instead of a single network improved the results significantly.

The features used to train the locating and classification tools were based on optical flow vectors. The flow vectors for consecutive frames were first measured. Based on statistical analysis and the anatomical shape of the body, six dynamic and adaptive channels were constructed. Principal component analysis was then applied to each channel to discard the correlated data for more accurate results. Experimental results clearly demonstrated the significant effect of using dynamic adaptive channels as a replacement for fixed size static channels where the proposed channels are adaptive to the perspective size and different pose of the body.

The results clearly show that while the temporal location of actions did not improve significantly when using dynamic channels, a 33.33% improvement in results for classification of human actions was observed when dynamic channels and neural networks in the form of anticipators (instead of one global network) were employed.

ACKNOWLEDGMENT

The authors would like acknowledge Paul Slinger for his aid through the course of this research.

REFERENCES

[1] N. Ikizler, G. R. Cinbis, and P. Duygulu, “Human action recognition with line and flow histograms”, in 19th IEEE International Conference on Pattern Recognition, pp. 1-4, 2008.

[2] A. Fathi and G. Mori, “Action recognition by learning mid-level motion features”, in IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8, 2008.

[3] S. Ali and M. Shah, “Human Action Recognition in Videos Using Kinematic Features and Multiple Instance Learning”, this paper appears in IEEE Transactions on Pattern Analysis and Machine Intelligence, Accepted for future publication, ISSN: 0162-8828.

[4] M. Ahmad and S. W. Lee, "Human Action Recognition Using Multi-View Image Sequences Features", in 7th IEEE International Conference on Automatic Face and Gesture Recognition, pp. 523-528, 2006.

[5] X. Li, “HMM based action recognition using oriented histograms of optical flow field” in IEEE Electronics Letters, Volume 43, Issue 10, pp. 560-561, 2007.

[6] L. Wang, “Abnormal Walking Gait Analysis Using Silhouette-Masked Flow Histograms”, 18th International Conference on Pattern Recognition, Vol. 3, pp. 473 – 476, 2006.

[7] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, “Recognizing Action at a Distance”, Proceedings of the 9th IEEE International Conference on Computer Vision, Vol. 2, pp. 726 - 733, 2003.

[8] G. Zhu, C. Xu, Q. Huang, and W. Gao, “Action Recognition in Broadcast Tennis Video”, 18th International Conference on Pattern Recognition, Vol. 1, pp. 251 – 254, 2006.

[9] T. Zhao and R. Nevatia, “3D tracking of human locomotion: a tracking as recognition approach”, Proceedings of the 16th International Conference on Pattern Recognition, Vol. 1, pp. 546 – 551, 2002.

[10] H. L. Zhu, P. Y. Du, and J. Xiang, “3D Motion Recognition based on Ensemble Learning”, 8th International Workshop on Image Analysis for Multimedia Interactive Services, pp. 28 – 28, 2007.

[11] S. Zhang, M. H. Ang, W. Xiao, and C. K. Tham, “Detection of activities for daily life surveillance: Eating and drinking”, 10th International Conference on e-health Networking, Applications and Services, pp. 171 – 176, 2008.

[12] J. Yamato, J. Ohya, K. Ishii, “Recognizing human action in time-sequential images using hidden Markov model”, Proceedings 1992 IEEE Conference on Computer Vision and Pattern Recognition, pp. 379 – 385, 1992.

[13] Y. C. Wu, H. S. Chen, W. J. Tsai, S. Y. Lee, and J. Y. Yu, “Human action recognition based on layered-HMM”, 2008 IEEE International Conference on Multimedia and Expo, pp. 1453 – 1456, 2008.

[14] X. Li and K. Fukui, “View Invariant Human Action Recognition Based on Factorization and HMMs”, IEICE Transactions on Information and Systems, Vol. E91-D, Issue 7, pp. 1848-1854, 2008.

[15] C. Schuldt, I. Laptev, and B. Caputo, “Recognizing Human Actions: A Local SVM Approach”, Proceedings of the 17th International Conference on Pattern Recognition, Vol. 3, pp. 32 – 36, 2004.

[16] A. Madabhushi and J. K. Aggarwal, “Using head movement to recognize activity”, Proceedings of the 15th International Conference on Pattern Recognition, Vol. 4, pp. 698 – 701, 2000.

[17] P. Kornprobst, T. Vieille, and I. K. Dimo, “Could early visual processes be sufficient to label motions?”, Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, Vol. 3, pp. 1687-1692, 2005.

[18] R. Venkatesh Babu and K. R. Ramakrishnan “Recognition of human actions using motion history information extracted from the compressed video”, Image and Vision Computing, Vol. 22, Issue 8, pp. 597 - 607, 2004.

[19] Y. Kuniyoshi and M. Shimozaki, “A self-organizing neural model for context-based action recognition”, First International IEEE EMBS Conference on Neural Engineering, pp. 442-445, 2003.

[20] M. Shimozaki and Y. Kuniyoshi, “Integration of spatial and temporal contexts for action recognition by self organizing neural networks”, Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 3, pp. 2385-2391, 2003.

[21] T. Theodoridis and H. Huosheng, “Action classification of 3D human models using dynamic ANNs for mobile robot surveillance”, 2007 International IEEE Conference on Robotics and Biomimetics, pp. 371-376, 2007.