Recognition of Aggressive Human Behavior Using Binary Local Motion Descriptors

Datong Chen, Ming-yu Chen, Can Gao, Alex Hauptmann, Howard Wactlar and Ashok Bharucha

School of Computer Science
Carnegie Mellon University

Abstract – Video surveillance is an alternative approach to staff or self-reporting that has the potential to detect and monitor aggressive behaviors more accurately. In this paper, we propose an automatic algorithm capable of recognizing aggressive behaviors from video records using local binary motion descriptors. The proposed algorithm may increase the accuracy for retrieving aggressive behaviors from video records, and thereby facilitates scientific inquiry into this low frequency but high impact phenomenon that eludes other measurement approaches .

Keywords: Surveillance video, behavior recognition, binary local motion descriptor

1 Introduction

Video-based human action recognition addresses the problem of classifying simple human behavior units from video scenes. The biggest classification challenge is the fact that observed video appearances for each human action contain large variances stemming from body poses, non-rigid body movements, camera angles, clothing textures, and lighting conditions. There are two main approaches to analyzing human motions and actions: model-based and appearance-based.

A model-based approach employs a kinemics model to represent the poses of body parts in each snapshot of body action. A recognition algorithm first aligns the kinemics model to the observed body appearance in each video frame and then codes the motion of the body parts with the model transformations. Most kinemics models are closely related to the physical structure of the human body. Akita [6] decomposed the human body into six parts: head, torso, arms and legs, and built a cone model with the six segments corresponding to counterparts in stick images. Hogg [7] used an elliptical cylinder model to describe human walking. Hidden Markov Model (HMM) was used to recognize tennis actions [8]. Yamato, *et al.* extract symbol sequence from image sequence and build HMM to model tennis actions. Bregler [9] further extended HMM to dynamic models which contain spatial and temporal blob information extracted from human bodies. Lee, *et al.* [19] applied a particle filter on a set of constraints on body poses. Finally, Deutscher, *et al.* [18] propose an annealed particle filter method that uses simulated annealing to improve the efficiency of searching. Model-based approaches require reliable analytical body parts detection and tracking, complex computer vision problems that merit further exploration.

An appearance-based method builds classifiers to directly remember the appearance of actions in each class without explicitly representing the kinemics of the human body. A good example is template matching, which is widely used as an appearance-based action recognition algorithm. Polana, *et al.* [10] computed a spatio-temporal motion magnitude template as the basis for activities recognition. Bobick, *et al.* [11] constructed Motion-Energy Images (MEI) and motion history images as temporal templates and then searched the same patterns in incoming test data. Appearance models can be generally extended to detect various actions without introducing knowledge on constructing domain specific models. However, appearance-based methods require more training examples to learn appearances under different body poses and motions compared with model-based methods. Many appearance-based methods also rely deeply on adequate actor segmentations that are difficult to guarantee.

In recent years, a branch of appearance-based approaches called *part-based approaches* is attracting interest. A part-based method decomposes the entire appearance of an actor into a set of small, local spatio-temporal components, and applies statistical models to map these local components to actions. It has adequate scalability and does not require constructing specific models as is the case with model-based approaches.. It is also more robust under varying translations, background noise, 2D rotations, and lighting changes than appearance-based methods that require global appearances. Local features in the space-time representation have been applied to human action recognition with an SVM classifier [28]. As an alternative, Dollár, *et al.* [13] proposed to detect sparse space-time interest points using linear filters. Niebles, *et al.* [24] considered an unsupervised learning method to categorize and localize human actions with a collection of spatial-temporal interest points. Ke, *et al.* [14] proposed volumetric features to describe events. The features are extracted from optical flow and are represented as combinations of small volumes.

We propose to characterize human behaviors in surveillance video though the use of spatio-temporal video cubes. A spatio-temporal video cube is a small, short and local video sequence extracted from an interest point to capture small but informative motions in the video. These small motions can be finger raising, knee bending, or lips moving. We assume that a behavior can

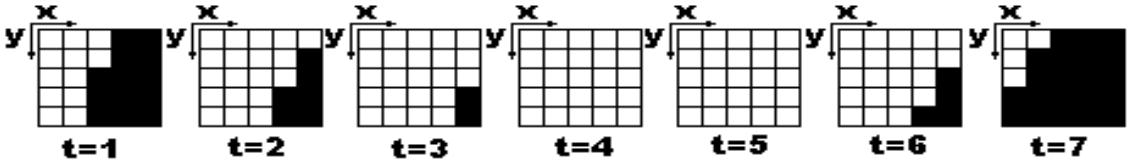

Figure 1. An illustration of local binary motion descriptors

be described by combination of these different types of movements. Since the extracted cubes are small, we believe they capture local appearance and are invariant to global appearance, posture, illumination, occlusion, etc. Thus, the fundamental problem of comparing the similarity of two behaviors becomes a search for similar, conceptually-meaningful components exhibited in the video.

Detection of Points of Interest

Local representations are usually extracted from certain interesting points instead of all the image pixels in a video. Typically, an interest point is extracted as a local response maxima pixel that corresponds to a predefined response function. In 2D images, such a response function could be a corner detector. In video, a spatio-temporal corner can be defined as a spatial corner that contains non-constant movements. Laptev, *et al.* [12] extended Harris interest point detector to extract spatial-temporal corners in a video sequence. Spatial-temporal corners are spatial interest points corresponding to the moments with non-constant motion. . In other words, a spatial-temporal corner is a region with strong local gradients in orthogonal directions along *x*, *y*, and *t*, i.e., a spatial corner or edge whose velocity vector is changing. In practice, true spatial-temporal corners are quite rare. This proved to be a challenge in detection and recognition tasks observed by Lowe [15]. Therefore, another local spatial-temporal interest point detector is proposed to detect periodic movements [13]. It applies 1D Gabor filters temporally and attempts to capture periodic motions. This provides a richer set of features, but it remains to be seen whether complex actions can be represented by periodic motions alone.

Our proposed interest point detection is based on the Harris corner detector. Instead of corner points in spatial positions, we extract points along edges with velocity vectors by simply replacing the 2nd moment gradient matrix with gradient magnitudes of *x*, *y*, and *t*. The goal is to find high contrast points both in space and time. This will identify the points which are along edges in a video image and contain velocity vectors. This will provide dense rather than sparse features from extended Harris detector. It also contains points with a range of different types of motions, not just periodic motions.

The proposed formula for interest point calculation is as follows:

$$L(x, y, t, \Sigma) = I(x, y, t) * g(0, \Sigma)$$
$$R(x, y, t) = \sqrt{\left(\frac{\partial L}{\partial x}\right)^2 + \left(\frac{\partial L}{\partial y}\right)^2 + \left(\frac{\partial L}{\partial t}\right)^2} \quad (1)$$

L denotes a smoothed video, which is computed by a convolution between the original video *I* and a Gaussian smoothing kernel *g*. To simplify the computation, we only keep the diagonal values of the covariance matrix Σ and use the variances in *x*, *y*, and *t* dimensions independently to control smoothing scales in space and temporal sequence. The response function *R* combines the magnitudes in the space and temporal dimensions. We calculate the response function value for each pixel and extract local-maxima pixels as interest points. The gradient over time performs a similar function as background subtraction to remove static background and preserve moving objects. We calculate approximate gradients with Sobel operators instead of true gradients to speed up the algorithm.

Local binary motion descriptor

At each interest point, a cube C_{xyt} is extracted which contains the spatio-temporally windowed pixel values in the video. The window size is normally set to contain most of the volume of data that contributed to the response function. We first convert the cube C_{xyt} to be binary cube B_{xyt} by thresholding pixels in the cube with one threshold τ. The threshold τ is determined by performing a class variance algorithm [22] on the first frame of the cube. We choose only the first frame to determine the threshold because an edge passes the center of the first frame of the cube according to our definition of the interest point. We assume that one side of the edge belongs to the actor's body and the other side belongs to the background. Most likely, the two sides contain similar number of pixels. We expect an adequate threshold to be found by solving a binary classification problem as in the class variance algorithm. In other frames, there may be a very unbalanced number of pixels between the body and background regions due to the actor's motion, where an adequate threshold may be difficult to guarantee.

One advantage of converting the cube to be binary is that a binary cube is robust under lighting changes. Figure 1. displays frames of a binary 5x5x7 cube. We can see that an object moves out of the cube window from the right bottom and moves back. This binary cube

contains local shape and motion information, though it may not represent the true motion of the object precisely because of aperture limitations.

A local binary feature *BF*(*S*,*M*) is computed from a binary cube B_{xyt}, which consists of shape feature *S* and motion feature *M*. The shape feature *S* is the first frame of the binary cube. We characterize this frame by modeling the "0" and "1" regions with two Gaussians cross the spatial dimensions respectively.

$$S = \left(\mu_x^0, \sigma_x^0, \mu_y^0, \sigma_y^0, \mu_x^1, \sigma_x^1, \mu_y^1, \sigma_y^1\right) \quad (2)$$

The motion feature is defined as a vector which records the motions of the geometric means of the "0" and "1" regions between frames.

$$M = \left(\Delta x_2, \Delta y_2, \ldots, \Delta x_t, \Delta y_t\right) \quad (3)$$

The first frame has no motion features. If one of the regions moves out of the cube at frame *t*, we record motion features in frame *i* frame *i+1* as "*NULL*".

Feature codebook

Local motion features extracted from the same body part contain similar motion information. Therefore we can cluster them to reduce the feature space into a fixed size of feature codebook.

We apply a modified Kmeans algorithm to performance this spatial-constraint clustering. The detail algorithm was proposed by [chen08]. It uses a graphic model to group local object appearance features into clusters under the constraint that spatially nearby local features should most likely to be grouped into the same cluster. We replace their 2D appearance features by the proposed local binary motion features and train the clusters with the EM process in the algorithm.

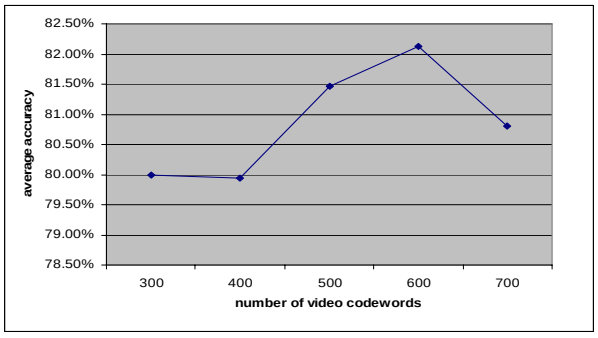

Figure 2: Classification performance using varying sizes codebooks in the KTH dataset.

The size of the codebook is determined by cross validation on the KTH human action dataset. KTH human motion dataset is widely used to evaluate event detection and recognition [4][12][13][14]. It's also the largest available video dataset of human actions for researchers to evaluate and compare with. The dataset contains six types of human actions (walking, jogging, running, boxing, hand waving, and hand clapping) performed by 25 different persons. Each person performs the same action four times under four different scenarios (outdoor, outdoor with different scale, outdoor with camera moving, and indoor). We performed leave-one-subject-out cross-validation to evaluate the size of the codebook. Figure 2 shows the recognition performance of using different sizes of video codebooks. The result shows there is a peak to achieve best performance (600 in KTH dataset). Too many video code words or too few video code words will all hurt the recognition performance.

Behavior classification

Human behaviors can vary greatly in global appearance. We may therefore extract a different number of video cubes from behavior sequences. This is a challenging problem in building behavior descriptor access by machine learning algorithms. The video codebook allows us to borrow the idea from document classification in building behavior descriptors. For each code in video codebook, we can treat it as a word in documents. In text classification, documents with different lengths are represented by a bag-of-words, which contains the frequencies of each word within a limited-size vocabulary. In our case, we can map extracted video cubes to their closest code word.

A behavior is represented by a histogram of all local binary features within a region of interest. The histogram is generated on the basis of the codebook, where code words are used as bins. Each local binary feature is mapped to its closest code word and added into the associated bin. We eventually normalize the counts in bins into frequencies. This descriptor does not consider the spatial correlations among local features, because the spatial information has somehow been used in the clustering step.

A behavior descriptor is treated as a vector with the same size as the codebook. Due to the rarity of aggressive behaviors in real life in comparison to normal behaviors, we use a one-center SVM to train a model for all normal behaviors and detect aggressive behaviors as outliers.

Experiments

We evaluate our algorithm using the CareMedia aggression dataset [5], that was collected from a real world surveillance video application. We demonstrate the robustness of our algorithm in recognizing aggressive behaviors in the CareMedia dataset. Forty-two physically aggressive behavior video clips and 1074 physically non-aggressive behavior video clips recorded in a dining room with multi camera views were labeled for training and testing. We used 1000 non-aggressive behavior video clips for training and the remaining 116 (42 + 74) clips for testing.

We smoothed input videos by a Gaussian filter with zero mean and variances (5, 5, 10) and extracted 5x5x10 video cubes from interest point. Each video cube was first converted into binary cube and then represented by local binary features. ROIs in the Caremedia dataset were labeled manually. We created a local binary behavior descriptor for each ROI in each video clip using a 600-word codebook.

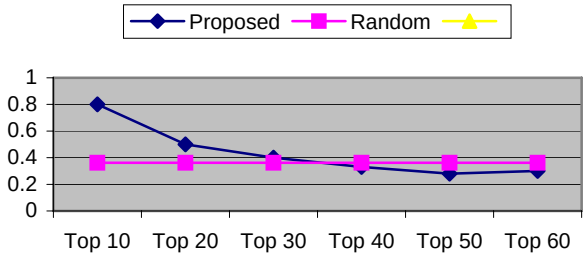

Figure 3. The aggressive behavior retrieval accuracy using the proposed method.

Figure 3 shows the performance of the proposed algorithm in recognizing aggressive behaviors. The top 10 retrieval results include about 80% aggressive behaviors, which is much better than the random accuracy 36.2%.

Figure 4. Examples of aggressive behaviors in the top 10 retrieval results.

Figure 5. Examples of aggressive behaviors in the last 20 retrieval results.

Figure 4 shows some frames extracted from the top 10 retrieval results. These behaviors involve large and colorful objects such as chairs and signs and can be well recognized by the proposed algorithm. Figure 5 shows some examples from the last 20 retrieval results. We can see that aggressive behaviors here are either occluded or only involve small objects that are difficult to notice even for humans. We also observed that many "aggressive behaviors" would not have been truly aggressive if they did not involve an object, i.e., spoon or chair. Recognizing subtle forms of aggressive behavior will require more than human kinemics models alone. Our approach is able to model the action of the arm, body, and the object together.

References

[1] Hu, W., Tan, T., Wang, L., and Maybank, S. A Survey on Visual Surveillance of Object Motion and Behaviors, IEEE Trans. SMC 3(34), Aug. 2004

[2] Fergus, R., Perona, P., and Zisserman, A. 2003, Object class recognition by unsupervised scale-invariant learning, *In CVPR*, 2003

[3] Agarwal, S., Awan, A., and Roth, D. 2004, Learning to detect objects in images via a sparse, part-based representation, PAMI, November 2004

[4] Schuldt, C. Laptev, I. and Caputo, B. 2004, Recognizing human actions: A local SVM approach. In ICPR, pages 32-36, 2004

[5] http://www.informedia.cs.cmu.edu/caremedia

[6] Akita, K. 1984, Image sequence analysis of real world human motion, Pattern Recognition, 17(1):73-83, 1984

[7] Hogg, D. 1983, Model-based vision: a program to see a walking person. Image and Vision Computing, 1(1):5-20, 1983

[8] Yamato, J., Ohya, J., and Ishii, K. 1992, Recognizing human action in time-sequential images using Hiden Markov Model, In CVPR, p. 379-385, Champaign, IL, June 1992

[9] Bregler, C. 1997, Learning and recognizing human dynamics in video sequences, In CVPR, San Juan, Puerto Rico, June 1997

[10] Polana, R., and Nelson, R. 1994, Low level recognition of human motion (or how to get your man without finding his body parts). In Proc. of IEEE Computer Society Workshop on Motion of Non-Rigid and Articulated Objects, p. 77-82, Austin TX, 1994

[11] Bobick, A.F. and Davis, J.W. 2001, The recognition of human movement using temporal templates. IEEE Trans. PAMI, 2001

[12] Laptev, I. and Lindeberg, T. 2003, Space-time interest points, In ICCV, p. 432-439, 2003

[13] Dollar, P., Rabaud, V, Gottrell, G. and Belongie, S. 2005. Behavior Recognition via Sparse Spatio-Temporal Features, In VS-PETS 2005, page 65-72

[14] Ke, Y., Sukthankar R., and Hebert, M. 2005, Efficient visual event detection using volumetric features. In ICCV, p. 166-173, 2005

[15] Lowe, D.G., 2004, Distinctive image features from scale invariant key points, IJCV, November 2004

[16] Yang, J., Jiang Y.G., Hauptmann, A. and Ngo, C.W. 2007, Evaluating bag-of-visual-word representation in scene classification, MIR'07 ACMMM, September 2007

[17] Basu, S., Bilenko, M., Banerjess, A. and Mooney, R.J. 2006, Probabilistic Semi-Supervised Clustering with Constraints, In Semi-Supervised Learning, MIT Press, 2006.

[18] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *IEEE CVPR*, volume 2, pages 126–133, 2000.

[19] M. W. Lee, I. Cohen, and S. K. Jung. Particle filter with analytical inference for human body tracking. In *IEEE Workshop on Motion and Video Computing*, pages 159–165, 2002.

[20] Schuldt, C., Laptev, I., and Caputo, B. Recognizing human actions: A local SVM approach. In ICPR, pp: 32–36, 2004.

[21] Niebles, J.C., Wang, H., Li,. F. Unsupervised learning of human action categories using spatial-temporal words. BMVC 2006.

[22] Wang, X., Wu, C. Approach of automatic multithreshold image segmation, in the 3rd World Congress on Intelligent Control and Automation, June 2000.