Human Action Recognition or Activity Recognition in Videos

Ajay Kumar¹, Rakesh Kumar Sharma², Sunil Kumar³
¹,²Electronics & Communication Engineering Department, Institute of Technology & Sciences, Bhiwani Haryana, INDIA
³Guide, Electronics & Communication Engineering Department, Institute of Technology & Sciences, Bhiwani Haryana, INDIA

ABSTRACT

This paper presents a novel approach to locate action objects in video and recognize their action types simultaneously using an associative memory model. The system uses a preprocessing procedure to extract key-frames from a video sequence and provide a compact representation for this video. Every training key-frame is partitioned into multiple overlapping patches in which image and motion features are extracted to generate an appearance-motion codebook. The training procedure also constructs a two-directional associative memory based on the learnt codebook to facilitate the system detecting and recognizing video action events using salient fragments, patch groups with common motion vectors. Our approach proposes the recently-developed Hough voting model as a framework for human action learning and memory. For each key-frame, the Hough voting framework employs Generalized Hough Transform (GHT) which constructs a graphical structure based on key-frame codewords to learn the mapping between action objects and a Hough space. To determine which patches explicitly represent an action object, the system detects salient fragments whose member patches are used to infer the associative memory and retrieve matched patches from the Hough model. These model patches are then used to locate the target action object and classify the action type simultaneously using a probabilistic Hough voting scheme. Results show that the proposed method gives good performance on several publicly available datasets in terms of detection accuracy and recognition rate.

Keywords---- component: Action shapes; Generalized Hough Transform; assiciation memory; salient fragment; human action detection and recognition

I. INTRODUCTION

To detect human action from a video sequence is a very important topic in computer vision due to its potential for many vision-based applications, such as video surveillance, man-machine interfaces, video indexing and retrieval, recognition of gestures, analysis of sports events, and authoring of video games [1]. Early works in action recognition focused on naming actions of single persons in controlled environments with simple and uniform background [2]. This limits their applications on unconstrained environments with complex motion, cluttered backgrounds, occlusions, illumination changes, viewpoint, pose, and geometric and photometric variances of objects. To solve these problems, various approaches typically employ machine learning tools to extract discriminative action features from video sequences, learn statistical action models, and classify action types of the input using the learned models [3].

In addition to classification, action recognition systems also try to detect and segment the observed motions into semantic meaningful instances of actions from videos [4,5]. To reach the goal, recent approaches consider the video action detection and recognition as an extension of 2D object detection [6-9] with higher dimensionality. Some of well-known approaches include space-time interest-point detectors and bag-of-words models [3,10]. These techniques aim at employing a combination of local space-time features and global 3D shape features to estimate the space-time boundaries of a given action. Two issues which are thereby of particular importance are to deal with local patch sampling and to explore the rich relationships among spatial-temporal "words" inherited from actions. While Yao et al. concluded in [3] that dense sampling would be superior to sparse sampling using an interest-point detector, they used fixed-size space-time patches which are not robust to temporal scaling in action detection and recognition.

The use of visual patterns in shape modeling is related to several ideas including the approach of local appearance codebooks [7,9] and the generalized Hough transform (GHT) [11] for object detection. At training time, these methods learn a model of the spatial occurrence distributions of local patches with respect to object centers. At test time, based on the trained classifier for each object class, the appearances of interest points in image or video are matched in the visual codebooks to detect a specific
object using the voting framework of GHT. The effectiveness of visual pattern grouping by Hough voting is thus well verified. A drawback of the approach of local appearance codebooks is that the time complexity to match patches with a large set of codebook entries during testing is high. The problem becomes serious when the approach is extended to detect human actions from videos due to the high dimensionality of the resulting codebook entries. To overcome this, Gall et al. [7] proposed a Hough forest approach for object detection that employs a tree-based approach to learn the patches in a supervised manner. Yao et al. [3] extended the method to address the problem of action detection and recognition. The time complexity of codeword matching for a patch is then proportional to the logarithm of the number of leaves in a Hough forest, which is efficient at runtime. The Hough forest approach attempts to construct a discriminative codebook for human action classification by incorporating offset uncertainty and class label uncertainty in the test of splitting leaf nodes which does not consider the resulting Hough voting space.

This paper presents an action detection and recognition system that represents highly variable action objects using mixtures of deformable patch models. The system represents an action object as a sequence of key-frames to tackle the temporal scaling problem in video action detection and recognition. The patches within bounding boxes for the objects in each key-frame are sampled to extract their spatial and temporal features for training these deformable patch models. Our approach builds on the GHT-voting framework. R-Tables in the GHT represent objects in each key-frame by a collection of patches arranged in a deformable configuration. We also address the problem of action detection and recognition based on only a salient fragment of original patches using an associative memory model that consists of the $R^{-1}$-tables defined by the inverse GHT and the deformable patch models.

The $R^{-1}$-table of an action class (or object) locates adjacent patches to a set of query patches and deformable patch models estimate the fidelities for these retrieved patches. A new set of query patches is then constructed and again used to search additional patches by applying the proposed associative memory modeling to improve the system performance. Our approach has the following contributions. Firstly, the approach is robust even when we apply it to detect video action objects from a video sequence with a cluttered background. One can recognize that not many stable patches can be sampled from a noisy video. Secondly, an associative memory approach is proposed to facilitate motion-vision cross-modal translation framework for video action detection and recognition. Thirdly, apart from the obvious manual initialization, in this paper, we propose an automatic training procedure for generating efficient and discriminative associative memory models [12] that sample descriptive model patches for usage in GHT-based human action detection and recognition. This tackles the critical initialization question with the usage of GHT which usually uses manually generated shape model as the central knowledge source. Finally, techniques to smartly select the patches simulating the target action shape are implemented to improve the trade-off between computational complexity of the GHT and the action detection accuracy. Results show that the proposed method gives good performance on several publicly available datasets in terms of detection accuracy and recognition rate.

II. THE GHT-BASED ACTION OBJECT REPRESENTATION

Fig. 1 shows the block diagram of the approach. A key-frame detection procedure is first applied to detect key-frames from a test video sequence. The system detects a key-object from each key-frame, piles up these key-objects to form the action object, and represents the action object by a patch-based GHT model. The GHT was initially proposed to represent an object with arbitrary shape using a so-called R-table [11]. We extend the R-table concept to represent an action object by a mixture of deformable patch models. A normalized action object can thus be described by the geometric relationship between the object reference point $X_{R}$ and centers $X_{s}$ of object patches using the GHT. Given a patch of center $X$, we can determine the reference point $X_{R}$ to be $X + d$, where $d$ is the offset vector from $X$ to $X_{R}$. Let $f_{i}$ be the feature vector extracted from patch $i$. We can perform a clustering algorithm on all patch feature vectors to obtain a codebook in which each codeword is the mean feature vector of the corresponding patch cluster. In this case, we call a patch cluster as a deformable patch model and the R-table is a mixture of deformable patch models.

Figure. 1. Block diagram of the proposed GHT-based human action detection and recognition using an associative memory model.

According to the R-table, we can perform the object search to locate the action object from frame images in a video sequence and recognize its action type simultaneously using patches’ information and a Hough
voting scheme. Considering a patch $P$ centered at $X=(x_P,y_P)$ in a frame of a test video sequence, $P$ would cast a vote on the cell $(x,y,s,c)$ of Hough parameter space $H$ where

$$
\begin{align*}
  x &= x_P + s_P x' \\
  y &= y_P + s_P y' \\
  s &= s_P \\
  c &= c_P
\end{align*}
$$

(1)

where $P$ is a matched patch in the R-table, $s_P$ is the size of $P$ ($P$), and $c_P$ is the class label of $P$. Then a voting process for increasing the value of $H(x,y,s,c)$ is executed. Multiple candidate patches can be activated from the $R$-table based on the feature vector of $P$. After all patches in a frame of a test video sequence have been processed, the 4D accumulator array will contain high peaks corresponding to reference point locations where instances of the learnt object occur in the frame. Moreover, we could construct the $R^2$-table built of a set of class inverting lists to reconstruct the target object when its reference point is determined.

We divide every 2D key-object in an action object into a set of overlapping 8x8 patches (blocks) which work as knowledge sources to represent the shape of the action object by the proposed GHT model. The features to characterize a patch including the histogram of gradient (HOG) [14] constructed from the edges within the patch, and the local motion vector. To learn the mapping between action objects and a Hough space, we follow the codebook approaches due to their greater flexibility and smaller number of training examples they need to see to learn implicit object shapes. Based on the features extracted from sampled patches, the step towards learning the appearance variability of an action class is to build up a visual codebook to characterize the member action objects. As the basic representation for our approach, for the collection of key-objects from all training action objects, we introduce the appearance-motion Hough model (AMHM), which is a two-directional codebook with two types of codewords, shown in Fig. 2. Obviously, the AMHM is a R-table in the proposed GHT voting framework.

The $AMHM$ is constructed from a set of patches

$$
\{P_i = \{HOG, MV\}, d_i\} \quad \text{where,}
$$

$P_i$ is a local patch sampled from a training action object; $P_i^{HOG}$ and $P_i^{MV}$ are the extracted patch features, the histogram of gradient (HOG) and motion vector (MV), respectively; $c_i$ is the action label ($c_i \in C$); $d_i$ is a 2D offset vector from the patch center to a key-shape center. The system uses an efficient $k$-means clustering [15] to cluster features of training patches regardless of their class labels. More concisely, each cluster of the $AMHM$ is a patch list which is indexed by two types of features, the HOG and MV codewords. The $AMHM$ is thus a two -directional Hough model which is described by the fusion of HOG and MV codebooks. The codewords for individual patch clusters are the corresponding deformable patch models. The number of clusters for the $AMHM$ model would be large because we need to derive many deformable patch models to describe the heavy patch variants in a large amount of training patches. This leads to a high computational complexity to search matched patches for a given query patch using the $AMHM$. To tackle the problem, our approach reduces the time complexity of patch searching using an associative memory ($AM$) model.

In the $AMHM$, let $m$ and $n$ are the numbers of the HOG and MV codewords, respectively. The $AMHM$ thus contains $mn$ patch clusters. The $AM$ model is a relation with three-dimensional tuples

$$
\{C_{ij}^{HOG}, C_{ij}^{MV}\}, \quad j = 1, \ldots, m, i = 1, \ldots, n
$$

where $w_{ij}$ is the weight of codeword pair $C_{ij}^{HOG}$, $C_{ij}^{MV}$. Generally, high-valued weight ($w_{ij}$), indicates a strong relationship between $C_{ij}$ codewords. We estimate the
value of $w_{ij}$ using the patches’ information in the patch cluster $PC_{ij}$ of the AMHM. Given a patch $P_k$ in $PC_{ij}$, the confidence to activate $P_k$ for Hough voting based on a feature type can be measured by

$$q(F_k, F_c) = \frac{1}{1 + D(F_k, F_c), D(F_k, F_c) = \|F_k - F_c\|}$$  \hspace{1cm} (2)

where $F_k$ and $F_c$ are the feature vectors of $P_k$ and codeword $c$, respectively; $D(, ,)$ returns the feature distance. We also define link connection $l_{ij}$ to indicate that patches in $PC_{ij}$ are activated by codewords $C_i^{HOG}$ and $C_j^{MV}$ simultaneously by a patch. The value of $l_{ij}$ is defined as

$$l_{ij} = \sum_{P_k \in PC_{ij}} q(F_k^{HOG}, C_i^{HOG}, F_k^{MV}, C_j^{MV}). \hspace{1cm} (3)$$

A sigmoid function is finally used to estimate the value of $w_{ij}$:

$$w_{ij} = Z \times \frac{1}{1 + \exp(-l_{ij}/\sigma^2)} \hspace{1cm} (4)$$

where $Z = 1/\sum_{i,j} w_{ij}$ is used to normalize the value of $w_{ij}$ over the range $[0,1]$ and $\sigma$ is a fixed value to determine the convergence speed of the weight. Actually, the value of $w_{ij}$ measures the accuracy of patch cluster $ij$ to be a deformable patch model. The size of the $AM$ is in general large; however, the entries of lower-valued weights could be eliminated from the $AM$ to reduce its dimension. That is, the system does not consider local patch deformable models of bad accuracy to simulate an action shape in the detection and recognition phase.

III. DETECTING AND CLASSIFYING ACTIONS BY INFERRING THE ASSOCIATIVE MEMORY

We first perform the proposed key-frame detection mentioned above to represent a test video $V$ as $k$ key-frames $V_1, V_2, \ldots, V_k$, where each of them is segmented into a set of assume that a test video sequence contains a single action shot boundary detection procedure into the proposed key-frame detection algorithm to make sure that a set of detected key-frames constitutes an action cycle.

The first step to classify and localize action shape from the test video sequence is the detection of salient fragments. To tackle the problem, we propose a Hough-based salient fragment detection algorithm to segment silent fragments.

Given a patch $P$ in current key-frame, we correlate $P$ with every patch $P'$ within a search window in previous key-frame

$$e(x, P', y) = \sum_{(x', y') \in P'} c_P(x, y) c_{P'}(x', y')$$

where $(d_x, dy)$ is the displacement vector between $P$ and $P'$, $c_P(x, y)$ and $c_{P'}(x', y')$ are the color vectors for the pixels at $(x, y)$ in $P$ and at $(x', y')$ in $P'$, respectively; $\|\|$ is the Euclidean distance, and $A = (d_x, dy)$ is the estimated motion vector between $P$ and $P'$. The similarity measure for patch pair $(P, P')$ is then obtained by

$$\xi = \frac{2}{1 + e^{-\delta_{(x', y')}^2}} \hspace{1cm} (6)$$

where $\delta$ is a positive constant, e.g., $\delta = 2$ to regulate the patch error. To detect salient fragments from key-frame $t$, the local similarity measurement for $(P, P')$ casts a vote on the 2D motion vector space $H_t(d_x, dy)$:

$$H_t(d_x, dy) = H_t(\delta) + \xi$$

Peaks induced by uniform patch matching would be spurious. To tackle the problem, we exclude uniform patches to cast votes on the parameter space because uniform patches in general produce in-accurate motion vectors. Many edge detectors such as the Canny edge operator can be used to detect edge points in each key-frame. In implementation, a patch with less 40% edge points is defined to be uniform.

Peaks in $H_t$ indicate salient fragments in key-frame $t$, whose member patches can be found by an inverse voting scheme. For these patches, we extract their HOG and motion feature vectors for further action detection and recognition based on the learnt $AM$ model. Notice that the motion vector of a patch in a salient fragment can be obtained from the corresponding peak parameters. We apply the well-known mean-shift mode estimation [16] on $H_t$ to detect salient fragments with significant peaks, and thus the followed action detection and recognision is fast since only a small amount of patches in key-frame $t$ are involved in the detected salient fragments.

Let $P \in F^{MV}$ be located at position $y \in R^2$ in key-frame $t$, where $F^{MV}$ and $F^{HOG}$ are the patch’s features, $c$ the patch’s unknown class label, and $sp$ is the size of $P$, be a patch from a detected salient fragment. The approach simila vecto entrie first searches $r$ motion $r$ s in the learnt associative memory $AM$ using $F^{MV}$. Let $P$ activate the
\[ j \]-th codeword in the \textit{AM} model. The patch information stored in cluster \( ij \) of the \textit{AMHM} model is accessed to conduct the job of action detection and recognition when the value of \( w_{ij} \) is larger than the average of the \( j \)-th column weights in \textit{AMAs} mentioned above, patches in a patch cluster of the \textit{AMHM} model are divided into multiple groups according to patches' offset vectors. Let \( L_{ij}^k \) be the \( k \)-th patch group of patch cluster \( ij \) in the \textit{AMHM} model. Given the set \( PC_P \) of patch clusters obtain by inferring \( P \) on the \textit{AM} model the spatial documentation action class and target object center in key-frame \( t \), can be decomposed by the following marginalization:

\[
F(\text{HOG}) = \sum_{c_{ij} \in C} \frac{H_{t}(c_{ij}, x_{c_{ij}})}{H_{t}(c_{ij}, x_{c_{ij}})} \tag{8}
\]

while the third term is approximated as the similarity measurement in terms of HOG feature vectors. We can then rewrite Eq. (9).

\[ F_k(\text{HOG}) \text{ and } y_k \text{ are the HOG feature vector and the location } \tag{10} \]

of the \( k \)-th patch sampled from a salient fragment in key-frame \( t \), respectively.

Performing the \textit{AMHM} vote generation algorithm, we can also compute the probability to label key-frame \( t \) as class \( c \) using the \textit{AMHM} \( M \):

\[
p(c \mid M, t) = \frac{H_{t}(c, x_{c})}{\sum_{c \in C} H_{t}(c, x_{c})} \tag{11}
\]

The rule to classify the input video sequence \( V \) which is represented by \( k \) key-frames is thus

\[
c(V) = \arg \max_{c \in C} (\sum_{t=2,\ldots,k} p(c \mid M, t)) \tag{12}
\]

Another approach is to extract the trajectory features based on the detected objects in key-frames. In some cases, it is impossible to distinguish an action from another relying only on the shape features. The classification accuracy could be further improved if we simultaneously use the shape features, i.e., the HOG feature vectors and the trajectory features derived by the proposed approach for video action classification.

IV. RESULTS

A series of experiments was conducted on an Intel PENTIUM Dual Processor 3.0GHz PC and three video datasets, the KTH dataset [17], the Weizmann dataset [7], and the UCF sport dataset [18], are constructed to evaluate the performance of the human action detection system. These datasets have been used in many human action recognition studies. First of all, we present some experimental results to illustrate the effectiveness of the proposed human action detection. Figure 3 shows the results of key-frames detection using two test video sequences from action classes ‘diving’ and ‘weight lifting’ in UCF. Fig. 4 shows the effectiveness of detecting salient fragments from a test video sequence ‘jogging’ in KTH. The Hough image \( H \) corresponding to the voting result of grouping patches into salient fragments between two consecutive key-frames shown in Fig. 4(a) is shown in Fig. 4(b). The peaks of \( H \) are obvious and easy to detect using a simple thresholding technique. Figs. 4(c) and 4(d) shows the detected key-objects using only patches in salient fragments and the result after inferring the associative memory, respectively. The results illustrate the effectiveness of the fragment-based approach for action object detection. Tables I and II show the comparison of localization results between the proposed method, other GHT-based method [3], and the vocabulary forest method.
All the compared methods perform well in action object detection and the proposed approach has the best performance in average detection accuracy. This illustrates the effectiveness of GHT-based method in video action object detection. However, our approach outperforms the compared methods for the input video sequences of cluttered backgrounds or occluded objects because the usage of salient fragments reduces many spurious votes in the resulting Hough images.

We further evaluated our system’s ability to apply the correct action label to a given video sequence and call this classification. Following the inspiration of [3], classification was measured with three variations of training and testing data: (A) training and testing on tracks generated from ground-truth annotations (B) training on tracks from ground truth and testing on automatically extracted tracks and (C) training and testing on automatically extracted tracks. We refer to these as data variations A, B, and C respectively from this point on. Results demonstrate that the proposed method is comparable to the method of Yao et al. [3] but fails in discriminating certain actions from others using information extracting only from key-objects. In the training database, video sequences might belong to different action classes but with similar sets of key-frames. In this case, shown in Fig. 5, the system generates multiple peaks in the resulting Hough image and thus is ambiguous in action classification. Our approach therefore incorporates a peak detection and verification module to eliminate false peaks from the Hough image generated by the Hough-voting framework using only the key-objects’ features. We implement the peak detection and verification procedure by the 3D template matching scheme proposed in our previous work [5], which uses features extracted from the whole action object of the input test sequence to classify all the details among ambiguous action classes. The postprocessing module improves the classification accuracy of the system by about 15% and thus the system is comparable with Yao et al.’s method and this further illustrates the effectiveness of Hough-voting framework in human action detection and recognition. Moreover, the proposed method can be considered as a coarse-to-fine approach in the temporal domain. Given an input video sequence, its key-frames (key-objects) are used to find out candidate classes and information extracted from the whole input action object is used to determine the correct class label from these candidate classes. The system is thus fast and accurate in human action classification using the proposed Hough-voting framework.

| Table I: KTH localization results. |
|------------------|------------------|
| Method           | Box   | Jog   | Run   | Walk  | Wave  | Average |
| Yao et al. [3]   | 0.88  | 0.95  | 0.84  | 0.72  | 0.95  | 0.95    |
| Our forest [4]   | 0.98  | 0.91  | 0.91  | 0.78  | 0.95  | 0.89    |
| Proposed         | 0.97  | 0.98  | 0.97  | 0.92  | 0.95  | 0.95    |

| Table II: UCF localization results. |
|------------------|------------------|
| Class            | Precision       |
| Yao et al. [3]   | Proposed        |
| Box              | 0.62            | 0.70            |
| Jog              | 0.78            | 0.59            |
| Run              | 0.37            | 0.65            |
| Walk             | 0.87            | 0.81            |
| Golf             | 0.77            | 0.51            |
| Sk. Board        | 0.39            | 0.42            |
| Average          | 0.67            | 0.73            |

V. CONCLUSION
In this paper we have presented a human action detection method based on detecting the salient fragments from two consecutive key-frames. Based on patches in the detected salient fragments, an associative memory inferring procedure is applied to locate action object and classify action types simultaneously from a test video sequence. Our approach inherited the robustness of the GHT voting framework in human action detection and recognition without the main disadvantage of high-computational complexity for traditional GHT. In our current work, our system shows promising performance using GHT to specify the spatial structure of an object; with more sophisticated local to global integration schemes, we expect the framework can be applied to other applications such as video event retrieval. The speed of detections can also be increased without affecting the robustness of the system by incorporating fast k-NNR searches into the framework. Future work will deal with adding complex event detection to the proposed system, and increasing the database size.
REFERENCES


