INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

Segmenting and Recognizing Human Action from Long Video Sequences

Ramya A.
Department of Computer Science & Engineering, Prist University, Puducherry, India
Backiyalakshmi R.
Department of Computer Science & Engineering, Prist University, Puducherry, India

Abstract:
Object detection and tracking in video is a challenging problem and recognizing human activities from video are one of the most promising applications of computer vision. In this paper, we present a general framework to jointly segmenting and recognizing videos of human action sequences. Hence, we use two approaches: (i) Intensity Range Based Background Subtraction (ii) Shape-Motion Prototype-Based approach. Here first one defines an intensity range for each pixel location in the background to accommodate illumination variation as well as motion in the background and second one is introduced for action recognition. It performs recognition efficiently via tree-based prototype matching and look-up table indexing. It captures correlations between different shape and motion by learning action prototypes in a joint feature space. It also ensures global temporal consistency by dynamic sequence alignment.

Key words: *Background modeling, background subtraction, video segmentation, Action recognition, shape-motion prototype tree, hierarchical K-means clustering*

1. Introduction

Action recognition is of central importance in computer vision with many applications in visual surveillance, human computer interaction and entertainment, among others. Object recognition is one of the core problems in computer vision, and it is a very extensively investigated topic. Applications such as video database, virtual reality interfaces, and smart surveillance systems all have in common tracking and interpreting human activities. Indoor surveillance provides information about areas such as building lobbies, hallways, and orcas. Monitoring in lobbies and hallways include detection of people depositing things (unattended luggage in an airport lounge), removing things (theft) or loitering. Outdoor surveillance includes tasks such as monitoring of a site for intrusion or threats from vehicle (e.g., car bombs). In these applications, people are the key element of the system. The ability to detect and track people with their body parts is therefore an important problem. In most of the suggested schemes, the object detected is accompanied with misclassified foreground objects due to illumination variation or motion in the background. The suggested background model initially determines the nature of each pixel as stationary or non-stationary and considers only the stationary pixels for background model formation. In the background model, for each pixel location a range of values is defined. Subsequently, in object extraction phase our scheme employs a local threshold, unlike the use of global threshold in conventional schemes. The main contribution of this paper is an approach for action recognition based on Shape-Motion Prototype-Based approach.

2. Related Works

Feature extraction methods for activity recognition can be roughly classified into four categories: geometry-based [4],[5], [6], motion-based [7], [8], [9], [10], appearance-based [4],[11], [12], and space-time feature-based [13], [14], [15].The geometry-based approaches recover information about human body configuration, but they often heavily rely on object segmentation and tracking, which is typically difficult and time consuming. The motion-based approaches extract optical flow features for recognition, but they rely on segmentation of the foreground for reducing effects of background flows. The appearance-based approaches use shape and contour information to identify actions, but they are vulnerable to cluttered complex backgrounds. The space time feature-based approaches either characterize actions using global space-time 3D volumes or more compactly using sparse space-time interest points.
From the existing literature, it is observed that most of the simple schemes are ineffective on videos with illumination variations, motion in the background, and dynamically textured indoor and outdoor environment, etc. Keeping this in view, we suggest here a

simple scheme called Local Illumination based Background Subtraction (LIBS) that models the background by defining an intensity range for each pixel location in the scene. Motivated by these issues, we introduce an efficient, prototype-based approach for action recognition. It's global temporal consistency by dynamic sequence alignment. In addition, it has the advantage of tolerating complex dynamic

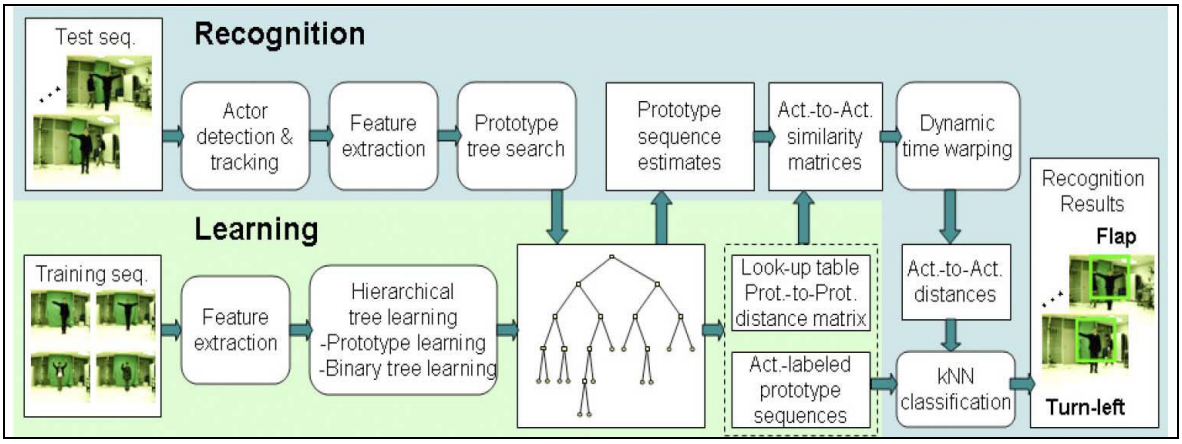

Figure 1: Overview of our approach

Backgrounds due to median-based background motion compensation and probabilistic frame-to prototype matching.

3. Proposed Scheme

The block diagram of our approach is shown in Fig. 1. Local Illumination based Background Subtraction (LIBS) that models the background by defining an intensity range for each pixel location in the scene. Subsequently, a local thresholding approach for object extraction is used. During training, an action prototype tree is learned in a joint shape and motion space via hierarchical K-means clustering and each training sequence is represented as a labeled prototype sequence; then a look-up table of prototype-to-prototype distances is generated. During testing, based on a joint probability model of the actor location and action prototype, the actor is tracked while a frame-to-prototype correspondence is established by maximizing the joint probability, which is efficiently performed by searching the learned prototype tree; then actions are recognized using dynamic prototype sequence matching. Distance measures used for sequence matching are rapidly obtained by look-up table indexing, which is an order of magnitude faster than brute-force computation of frame-to-frame distances.

3.1. LIBS Scheme

The LIBS scheme consists of two stages. The first stage deals with finding the stationary pixels in the frames required for background modeling, followed by defining the intensity range from those pixels. In the second stage a local threshold based background subtraction method tries to find the objects by comparing the frames with the established background.

LIBS uses two parameters, namely, window size (an odd length window) and a constant for its computation. The optimal values are selected experimentally. Both stages of LIBS scheme are described as follows.

Algorithm 1 Development of Background Model

Consider n initial frames as $\{f_1, f_2, \cdots, f_n\}$, where $20 \leq n \leq 30$.
for $k \leftarrow 1$ to $n - (W - 1)$ **do**
  for $i \leftarrow 1$ to height of frame **do**
    for $j \leftarrow 1$ to width of frame **do**
      $\overrightarrow{V} \leftarrow [f_k(i,j), f_{k+1}(i,j), \ldots, f_{k+(W-1)}(i,j)]$
      $\sigma \leftarrow$ standard deviation of $\overrightarrow{V}$
      $\mathbf{D}(p) \leftarrow |V(k + (\lfloor W \div 2 \rfloor)) - V(p)|$, for each value of $p = k + l$, where $l = 0, \cdots, (W - 1)$ and $l \neq \lfloor W \div 2 \rfloor$
      $\mathbf{S} \leftarrow$ sum of lowest $\lfloor W \div 2 \rfloor$ values in $\overrightarrow{\mathbf{D}}$
      if $\mathbf{S} \leq \lfloor W \div 2 \rfloor \times \sigma$ **then**
        Label $f_{k+(\lfloor W \div 2 \rfloor)}(i,j)$ as stationary
      else
        Label $f_{k+(\lfloor W \div 2 \rfloor)}(i,j)$ as non-stationary
      end if
    end for
  end for
end for
for $i \leftarrow 1$ to height of frame **do**
  for $j \leftarrow 1$ to width of frame **do**
    $M(i,j) = \min[f_s(i,j)]$ and $N(i,j) = \max[f_s(i,j)]$, where $s = \lceil W \div 2 \rceil, \cdots, n - (\lfloor W \div 2 \rfloor)$ and $f_s(i,j)$ is stationary
  end for
end for

3.1.1. Development of Background Model

Conventionally, the first frame or a combination of first few frames is considered as the background model. However, this model is susceptible to illumination variation, dynamic objects in the background, and also to small changes in the background like waving of leaves etc.

3.1.2. Extraction of Foreground Object

After successfully developing the background model, a local thresholding based background subtraction is used to find the foreground objects. A constant is considered that helps in computing the local lower threshold and the local upper threshold. These local thresholds help in successful detection of objects suppressing shadows if any. The steps of the Algorithm are outlined in Algorithm 2.

Algorithm 2 Background Subtraction for a frame f

for $i \leftarrow 1$ to height of frame **do**
  for $j \leftarrow 1$ to width of frame **do**
    Threshold $\mathbf{T}(i,j) = (1/C)(M(i,j) + N(i,j))$
    $\mathbf{T_L}(i,j) = M(i,j) - \mathbf{T}(i,j)$
    $\mathbf{T_U}(i,j) = N(i,j) + \mathbf{T}(i,j)$
    if $\mathbf{T_L}(i,j) \leq f(i,j) \leq \mathbf{T_U}(i,j)$ **then**
      $\mathbf{S}_f(i,j) = 0$ //Background pixel
    else
      $\mathbf{S}_f(i,j) = 1$ //Foreground pixel
    end if
  end for
end for

3.2. Shape-Motion Prototype

During training, action interest regions are first localized and shape-motion descriptors are computed from them. Next, action prototypes are learned as the cluster centers of K-means clustering, and each training sequence is mapped to a sequence of learned prototypes. Finally, a binary prototype tree is constructed via hierarchical K-means clustering using the set of learned action prototypes. In the binary tree, each leaf node corresponds to a prototype.During testing, humans are first detected and tracked using appearance information, and a frame-to-prototype correspondence is established by maximizing a joint probability of the actor location and action prototype. Given the rough location of the actor by appearance-based tracking, joint optimization is performed to refine the location of the actor and identify the corresponding prototype. Then, actions are recognized based on dynamic prototype sequence matching. Distances needed for matching are rapidly obtained by look-up table indexing, which is an order of magnitude faster than the brute-force computation of frame-to-frame distances. Our main contributions are threefold: A prototype-based approach is introduced for robustly detecting and matching prototypes, and recognizing actions against dynamic backgrounds. Actions are modeled by learning a prototype tree in a joint shape-motion space via hierarchical K-means clustering. Frame-to-frame distances are rapidly estimated via fast prototype tree search and look-up table indexing.

3.2.1. Shape-Motion Descriptor

A shape descriptor for an action interest region is represented as a feature vector by dividing the action interest region into ns square grids (or subregions) R1 . . .Rns . Given the shape observations from background subtraction (when the camera and background are static) or from appearance likelihood maps (for dynamic cameras and background).

$$D_s = (s_1 \ldots s_{n_s}) \in \mathcal{R}^{n_s}$$

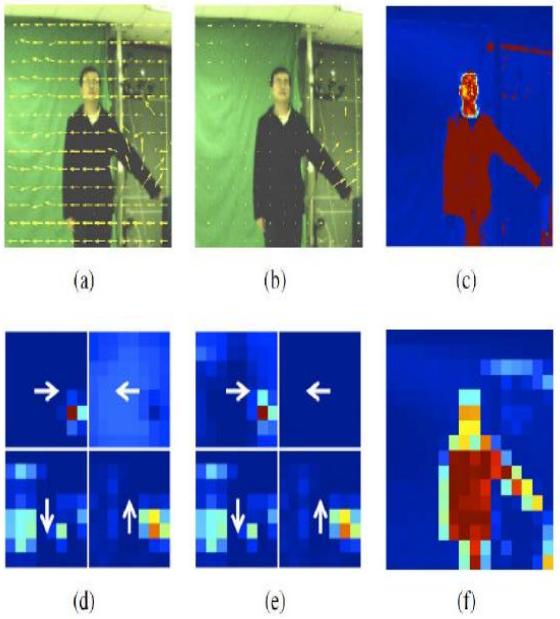

Fig. 3. An example of computing the shape-motion descriptor of a gesture frame with a dynamic background. (a) Raw optical flow field, (b) compensated optical flow field, (c) combined appearance likelihood map, (d) motion descriptor D_m computed from the raw optical flow field, (e) motion descriptor D_m computed from the compensated optical flow field, (f) shape descriptor D_s. A motion descriptor is visualized by placing its four channels in a 2×2 grid.

Figure 3

3.2.2. Shape-Motion Prototype Tree

Given the set of shape-motion descriptors for all frames of the training set, we perform K-means clustering in the joint shape-motion space using the Euclidean distance for learning the action prototypes. The cluster centers are then used as the action prototypes. In order to rapidly construct frame-to-prototype correspondence similar to the online matching of shape exemplar by tree traversal we

build a binary prototype tree over the set of prototypes based on hierarchical K-means clustering and traverse the tree to find the nearest neighbor prototype for any given test frame (i.e., observation V) and hypothetical actor location, during testing.
During tree construction, an initial 2-means clustering process is run on the action prototypes to partition the entire set of prototypes into two groups. Then, the same procedure is applied recursively to each group. This process will generate a quantization tree for finding nearest prototypes (leaf nodes). During testing, each query descriptor is passed down the tree by comparing the two candidate cluster centers at each level and then choosing the closest one. This matching process continues until it arrives at a leaf node. Examples of action prototypes and a binary prototype tree are shown in Fig. 4. We construct a prototype-to-prototype (pairs of leaf nodes) distance matrix which is computed offline in the training phase, and use it as a lookup table to speed up the action recognition process.

3.3. Action Recognition
The action recognition process is divided into two steps: frame-to-prototype matching and prototype-based sequence matching.
The conditional probability is decomposed into a prototype matching term (prototype likelihood given the actor location) and an actor localization term:

$$p(\theta, \alpha|V) = p(\theta|V, \alpha)p(\alpha|V). \quad (1)$$

Based on the tracking information, the actor localization term is modeled as follows:

$$p(\alpha|V) \propto \frac{L(\alpha|V) - L_{min}}{L_{max} - L_{min}}, \quad (2)$$

We model a prototype matching term as

$$p(\theta|V, \alpha) \propto e^{-d(D(V,\alpha),D(\theta))}, \quad (3)$$

Where d represents the Euclidean distance between the Descriptor determined by observation V at location, and the descriptor of the prototype.

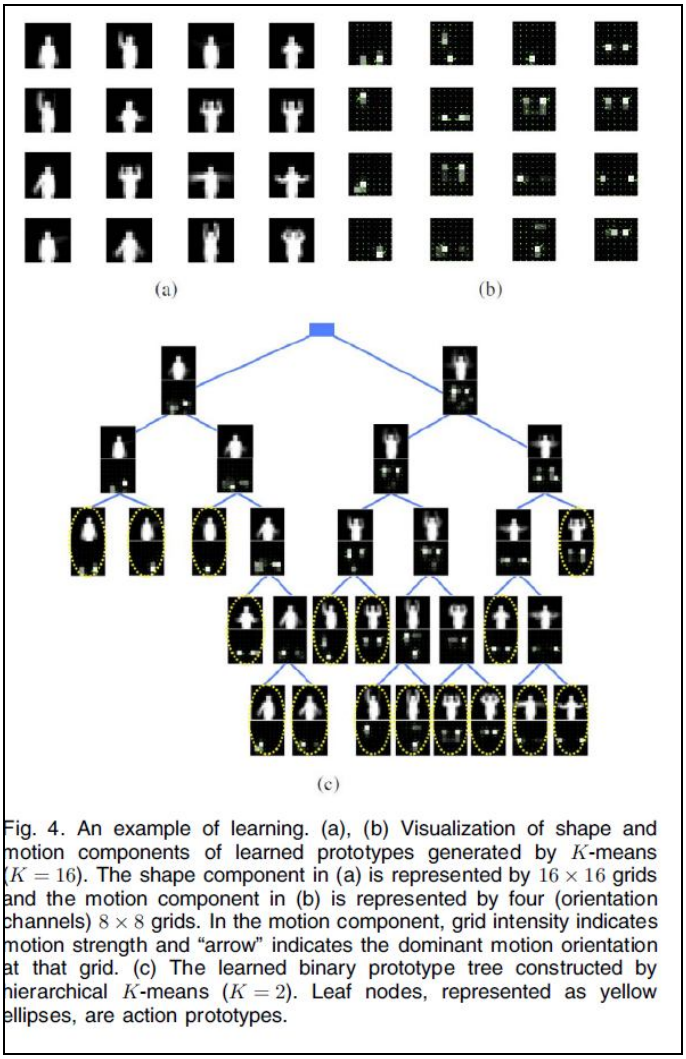

Fig. 4. An example of learning. (a), (b) Visualization of shape and motion components of learned prototypes generated by K-means ($K = 16$). The shape component in (a) is represented by 16×16 grids and the motion component in (b) is represented by four (orientation channels) 8×8 grids. In the motion component, grid intensity indicates motion strength and "arrow" indicates the dominant motion orientation at that grid. (c) The learned binary prototype tree constructed by hierarchical K-means ($K = 2$). Leaf nodes, represented as yellow ellipses, are action prototypes.

Figure 4

4. Conclusion

In this work we have proposed a simple but robust scheme of background modeling and local threshold based object detection. In general, it is observed that the suggested scheme outperforms others and detects objects in all possible scenarios considered. Our approach is both accurate and efficient for action recognition, even when the action is viewed by a moving camera and against a possibly dynamic background. This good performance is mostly due to the fact that our approach captures correlations between shape and motion by learning action prototypes in the joint feature space, and secondarily because it ensures global temporal consistency by dynamic sequence alignment.

5. References

1. T.B. Moeslund, A. Hilton, and V. Kruger, "A Survey of Adances in Vision-Based Human Motion Capture and Analysis," Computer Vision and Image Understanding, vol. 104, no. 2, pp. 90-126, 2006.
2. P. Turaga, R. Chellappa, V.S. Subrahmanian, and O. Udrea, "Machine Recogntion of Human Activities: A Survey," IEEE Trans. Circuits and Systems for Video Technology, vol. 11, no. 8, pp. 1473-1488, Nov. 2008.
3. R. Poppe, "A Survey on Vision-Based Human Action Recognition,"Image and Vision Computing, vol. 28, no. 6, pp. 976-990, 2010.
4. H. Li and M. Greenspan, "Multi-Scale Gesture Recognition from Time-Varying Contours," Proc. IEEE Int'l Conf. Computer Vision,vol. 1, pp. 236-243, 2005.
5. Y. Shen and H. Foroosh, "View-Invariant Action Recognition Using Fundamental Ratios," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-6, 2008.
6. P. Natarajan, V. Singh, and R. Nevatia, "Learning 3D Action Models from a Few 2D Videos for View Invariant Action Recognition," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-8, 2010.

7. A. Efros, A. Berg, G. Mori, and J. Malik, "Recognizing Action at aDistance," Proc. IEEE Int'l Conf. Computer Vision, vol. 2, pp. 726- 733, 2003.
8. G.R. Bradski and J.W. Davis, "Motion Segmentation and Pose Recognition with Motion History Gradients," Machine Vision and Applications, vol. 13, pp. 174-184, 2002.
9. A. Fathi and G. Mori, "Action Recognition by Learning Mid-Level Motion Features," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-8, 2008.
10. Y. Wang, P. Sabzmeydani, and G. Mori, "Semi-Latent Dirichlet Allocation: A Hierarchical Model for Human Action Recognition," Proc. IEEE Int'l Conf. Computer Vision Workshop Human Motion Understanding, Modeling, Capture and Animation, pp. 240-254, 2007.
11. A. Elgammal, V. Shet, Y. Yacoob, and L.S. Davis, "Learning Dynamics for Exemplar-Based Gesture Recognition," Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 571-578, 2003.
12. C. Thurau and V. Hlavac, "Pose Primitive Based Human Action Recognition in Videos or Still Images," Proc. IEEE Conf. ComputerVision and Pattern Recognition, pp. 1-8, 2008.
13. C. Schuldt, I. Laptev, and B. Caputo, "Recognizing Human Actions: A Local SVM Approach," Proc. Int'l Conf. Pattern Recognition, vol. 3, pp. 32-36, 2004.
14. I. Laptev and P. Perez, "Retrieving Actions in Movies," Proc. IEEE Int'l Conf. Computer Vision, pp. 1-8, 2007.
15. I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning Realistic Human Actions from Movies," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 1-8, 2008