Incremental Activity Modeling and Recognition in Streaming Videos
Mahmudul Hasan and Amit K. Roy-Chowdhury

Introduction

Most of the state-of-the-art approaches to human activity recognition in video are based on one or more of the following four assumptions:

1. Requires an intensive training phase, where every training example is assumed to be available.
2. Every training example is assumed to be labeled.
3. At least one example of every activity class is assumed to be seen beforehand.
4. A video clip contains only one activity, where the exact spatio-temporal extent of the activity is known.

However, these assumptions are too strong and not realistic in many real-world scenarios such as streaming and surveillance videos, where new unlabeled activities are coming continuously and the spatio-temporal extent of these activities are usually known in advance.

Goal of this Work

1. Classify new unknown activities in streaming videos.
2. And also leverage upon them to continuously improve the existing activity recognition system.

Framework and Overview

[Activity Segmentation Example Large] [Activity Classifiers] [Activity Segmentation Example Small] [Model Update] [Activity Learning System] [Activity Localization] [Teacher Selection] [Features] [Activity Classifiers]

[Activity Segmentation Example] [Activity Classifiers] [Activity Segmentation Example Small] [Model Update] [Activity Learning System] [Activity Localization] [Teacher Selection] [Features]

Performance of incremental learning framework in classifying some individual activities

(Images) Above illustrated actions are as follows (left to right, top to bottom): jogging, walking, handclapping, running, boxing, golf swing, diving, tennis, biking, soccer juggling, faceting out, and vehicle id. In blue line means correct, red line means negative, and white line means misclassification of the action at that particular instant.

Activity Model

We use an ensemble of multi-class linear SVM for activity modeling, which is defined as:

\[ H(x) = \sum \log(h_i(x)), \quad \text{where } h_i(x) \text{ is the } i^{\text{th}} \text{ classifier in the ensemble, } \hat{b}_i = \frac{1}{\gamma}, \text{ and } \gamma \text{ is the corresponding weight, and } \Psi \text{ is the normalized error due to } h_i. \]

Active Learning System

[Intuition] Each new classifier added to the ensemble is trained using a set of examples drawn according to a distribution from the buffer, which ensures that examples from activities that are currently being classified by all recent ensemble have high probability of being sampled in the next round.

[Teacher] They are mainly classification algorithms that make errors but perform above the accuracy of random guessing.

Overall Algorithm

1. Step 1: The model is trained using a batch of new examples (X, Y).
2. \( t \) is the total number of segmented rounds.
3. Step 2: \( H(t) \) is returned to the distribution, get new data, add to the buffer, then repeat Steps 1 and 2.
4. Step 3: Variable is updated over time. 

Experiments

[XTH Dataset] Six activity classes. Activity segmentations are given.

[UCF11 Dataset] Eleven activity classes. Activity segmentations are given.

Summary

Our framework is able to continuously improve the performance of the activity models using different feature sets; it achieves performance similar to the batch methods; it does not require storage of all training examples.

Acknowledgement

This work was supported in part by ONR grant N00014-12-1-0106 and NSF grant IIS-1316934.