Violent Crowd Action Recognition

Rubba Ashwas
University of Central Florida
rubba@knights.ucf.edu

Salman Khokhar
University of Central Florida
slmnkhokhar@gmail.com

Abstract

In our paper we aim to explore action recognition and test our implementation on our own dataset involving violent crowds and actions. We extracted features using the IDTF method and generation of MBH, HOF, and HOG as our descriptors for each input video tested. Then, using SVM training, we were able to perform classification for each action class, producing results. This has real world applications in areas such as video surveillance and security.

Keywords—action recognition; crowds; violent; IDTF

I. Introduction

Action detection still remains a largely unsolved problem. It is even more difficult to solve action detection in a crowd setting. The challenge in detecting and localizing these specific actions in a crowd scene includes: background activity, camera motion, and occlusions created by random objects in the scene (such as trees and buildings).

Using computer vision to recognize and detect specific actions in a scene can be useful for video surveillance applications. In violent crowds specifically, detecting the people engaging in violent actions would assist in locating the cause of the problem and then attempting to minimize it from there.

II. Related Work

In recent publications, the topic of action recognition has been explored in a non-crowd setting. Specifically, the implementation of removing inconsistent trajectories through the estimation of camera motion has allowed for the improved dense trajectory features (IDTF) to achieve current state-of the art results [1]. In Wang et al's paper, results showed that combining all descriptors increased overall performance and that issues arose when humans occupied larger areas in the foreground image [1].

Not much work has been done involving action detection in a crowd, so in Siva et al's paper the challenge is tackled by using a greedy k nearest neighbor algorithm to reduce manual annotations which showed comparable results against existing methods [2]. Fully annotated sequences helps improve temporal localization and tracking, which in turn makes SVM training more robust.

III. Violent Action Recognition Pipeline



Figure 1

We followed a standard action recognition pipeline for our experiments as presented in Figure 1. Features were computed on the entire video sequence and then collected in bounding boxes for which we generated a bag of words representation. An SVM classifier was trained to perform classification using negative and positive examples. All of the experiments were run on Matlab.

Features used: Improved dense trajectory features [1] were computed on the videos to capture action dynamics. IDTF computes dense trajectories along discriminative image points and computes Histograms of Orientated Gradients (HOG), Histograms of Optical Flow (HOF) and Motion Boundary Histograms (MBH) features along these trajectories. HOG captures shape information, HOF captures motion information and MBH captures relative motion.

Representation: To capture the features in each bounding box we used a bag of words representation. A dictionary is learned over the features using simple kmeans. The features in each box are assigned to these dictionary elements and a histogram is generated from these assignments. Each box is then represented by one histogram.

Classifier learning: To perform classification, we employed a simple SVM classifier. Training and testing is performed using a leave-one-out framework over the people in a video. SVM paramaters were tuned using a grid search. Additionally, we used an RBF kernel.

IV. Dataset

For our experiments and SVM training, we collected our own dataset in order to provide lots of relevant sequences involving violent crowds and to classify specific actions that we wanted to detect. In total, 32 video clips were collected and manually annotated, which include human actions in a crowd or a large group setting. Each clip was around 150-400 frames long, containing 20-100 people in each scene. On average, each frame sequence was annotated with a total of about 20-50 people tracked along each scene, including those performing the specific action and those not. A box was drawn on each person and tracked along the sequence, skipping every 10 frames. For all annotations, the naming convention was the same, tList_Object_###.mat, in which the number reset for each frame sequence being annotated

Once annotations were complete, they were manually labeled on whether the annotation at the current frame included the present action being looked at. These were labeled isAction_Object_###.mat where 'Action' was replaced to specify the class it was part of, such as isFalling_Object_001.mat.

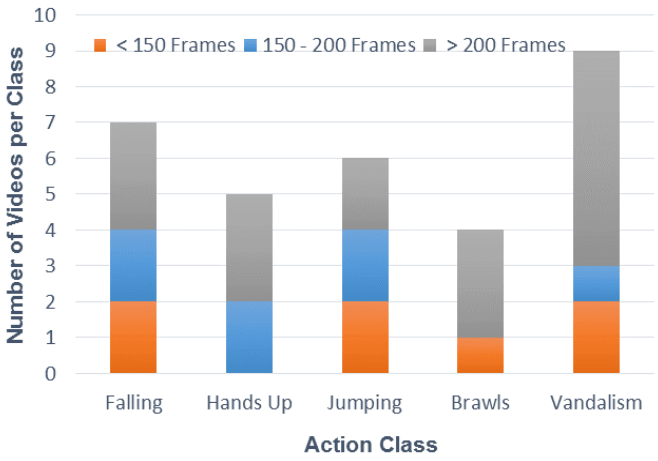


Figure 2: This chart represents the statistics on the amount of frames in each sequence per video in each Action class.

Action Classes: The dataset includes a total number of five action classes: Falling, Hands Up, Jumping on Objects, Brawls, and Vandalism. All these videos were collected from YouTube. Their categories are grouped in Figure 3 and their statistics are shown in Figure 2.



Figure 3: Our crowd action data set, where 34 scenes are available for testing from the 32 videos collected. 31 of these videos were fully annotated in correspondence to the action class it is part of.



Figure 4: This is an example frame from one of the videos in the Falling action class, of the manual annotations made. Thicker lines represent the current action being looked at, in this case, falling. The thinly-lined boxes indicate negative examples.

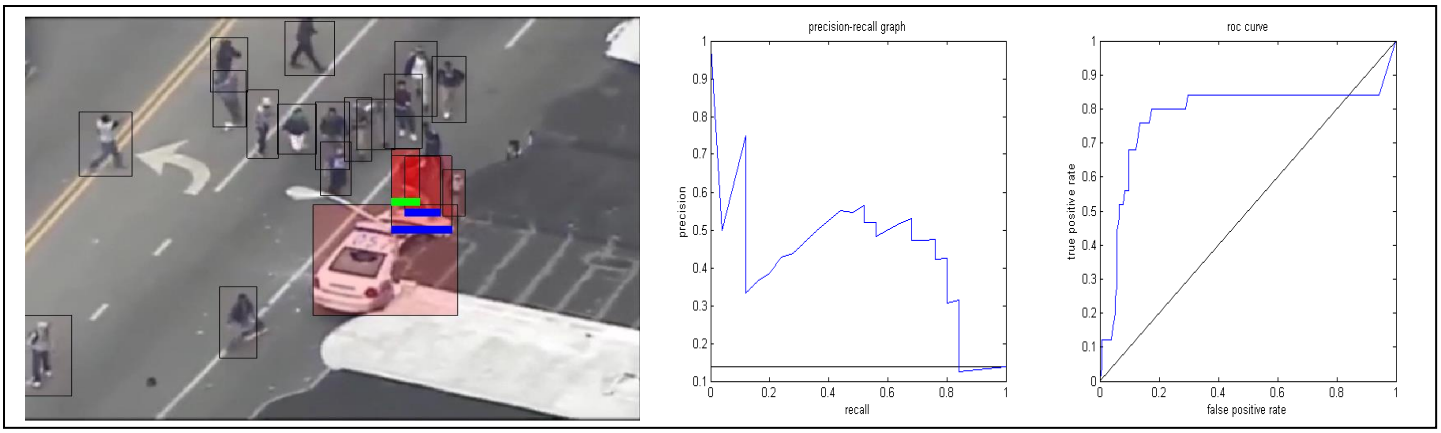
V. Results

We tested 14 videos from our dataset on our SVM training algorithm and received comparable results on the precision-recall graphs and roc curves.

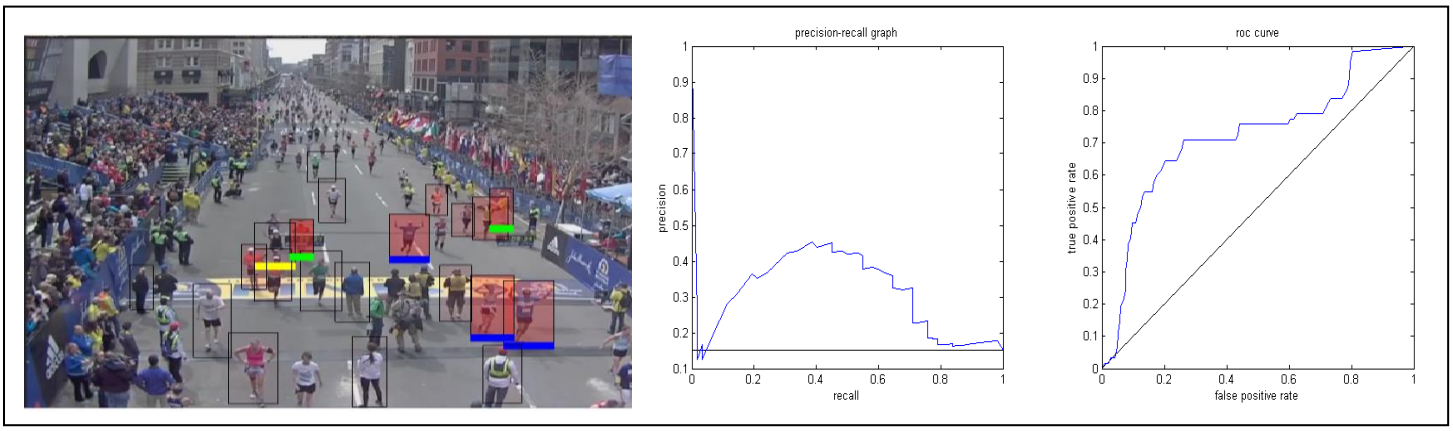
Color Codes: The colored bars at the bottom of each annotation box represent different results of SVM training. Yellow bars represent missed detections, green bars represent false positives, and blue bars represent correct detections. The amount of red fill in each box corresponds to the confidence level of each detection. No fill indicates that the person inside the box is not performing the action being looked at.



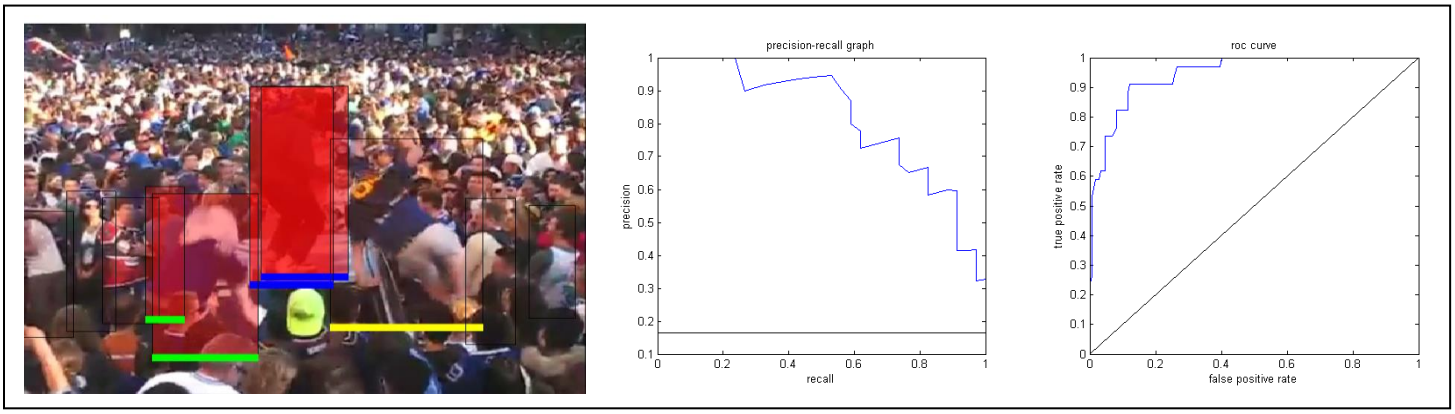
(a)



(b)



(c)



(d)

Figure 5: SVM results and corresponding graphs. **(a)** London Riots [Vandalism Class] - 37% and 87% **(b)** Baltimore Riots [Jumping Class] - 41% and 78% **(c)** Boston Marathon Bombing [Hands Up Class] – 30% and 72% **(d)** Vancouver Riots [Jumping Class] – 76% and 95%

VI. Conclusion

Overall, the algorithm used in experimentation was moderately effective, as there were still issues identifying actions and misidentifications of certain people in each scene. Low image quality and camera motion made some videos difficult to identify the humans in that scene. Future work would include minimizing missed detections by improving human detection and stabilization of the videos.

References

[1] Heng Wang, Cordelia Schmid. Action Recognition with Improved Trajectories. ICCV 2013 - IEEE International Conference on Computer Vision, Dec 2013, Sydney, Australia. IEEE, pp.3551-3558, 2013

[2] P. Siva and T. Xiang. Action detection in crowds. In BMVC, 2010