Recognizing human actions in videos

Praneeth A S
UG201110023
B.Tech (IV year)
Computer Science & Engineering

Anuroop Kakkirala
UG201113017
B.Tech (IV year)
Systems Science

Mentor: Dr. Gaurav Harit
Assisant Professor
Dept. of Computer Science & Engineering

Indian Institute of Technology Jodhpur
Jodhpur, Rajasthan 342011, India

April 29, 2015

Contents

List of Algorithms

List of Figures

1 Abstract

A **video** is a moving sequence of images at a particular frame rate. Unless if a video is created artificially, video captured by still cameras mostly has some or other kind of an object moving. Moving objects usually can be described by some **action words**. It is easy for humans to recognize which object is performing which action, but it is not possible for machines. But each recognition task whether recognizing humans or their actions would require some processing on the video. We have aimed to do some processing on images and videos to obtain some features which could be useful in recognition. In the first semester, we have tried using Histogram of Oriented Gradients to capture certain features specific to humans. These features were then trained using SVM for recognition. Im the second semester we used space time interest points [1] to gather HoG features [2]. These features were clustered using k-means which where then trained using multi-svm classifier. There has been a huge accuracy (∼80%) in recognizing human actions.

2 Phase 1

2.1 Introduction

Detecting humans in images is a challenging task owing to their variable appearance and the wide range of poses that they can adopt. The first need is a robust feature set that allows the human form to be discriminated cleanly, even in cluttered backgrounds under difficult illumination. We study the issue of feature sets for human detection, showing that locally normalized Histogram of Oriented Gradient (HOG) descriptors. For simplicity and speed, we use linear SVM as a baseline classifier throughout the project.

2.2 Algorithm

Algorithm 2.1 Histogram of Oriented Gradients

Input: Image of size 128×64

Output: Histogram of Oriented Gradients descriptor for the image

1. Divide the Image window of size 128×64 into 8×8 blocks // Total 15×7 blocks
2. Add a 1px border of zeros to the image.
3. Take Horizontal(dx) and Vertical Gradient(dy) of the image
4. Derive the angle and magnitude matrix using dx and dy
5. For every block quantize the gradient orientation into 9 bins based on magnitude
6. Apply normalization procedure on obtained gradients and concatenate the obtained histograms // Size of HoG = 15*7*4*9=3780

2.3 Experimental Setup

2.3.1 Dataset

We have used INRIA dataset [3] which were created in 2005 by Navneet Dalal. The data set has been divided into two sets: one for training and one for testing. There were 614 images of humans in different backgrounds & 1218 images of non-humans like buildings, vehicles etc., in training set. There were 228 images of humans in different backgrounds & 453 images of non-humans like buildings, vehicles etc., in testing set. The humans in both test and training set were in different poses with different backgrounds.

2.3.2 Experiment

The images were scaled to 128×64 before they were used for any processing. We have performed the training & testing with both colour and grayscale images.First, we have computed gradient images using various filters like centered, uncentered and cubic centered. We did this to ensure normalized colour and gamma values. Then we computed histograms of images with 8 and 9 number of orientations in two modes: 'signed'(0 to 2π) & 'unsigned'(0 to π). 9 bins had each bin with 20° in unsigned mode. For the block normalization, we have used two schemes L1-norm and L2-norm.

$$\text{L2-norm } \nu = \frac{\nu}{\sqrt{||\nu^2 + \epsilon^2||}} \text{ \& L1-norm } \nu = \frac{\nu}{||\nu + \epsilon||}$$

The final step was to train the obtained features using Linear SVM. We've used SVM Light [4] package for the same. The training positive features were labelled +1 and negative features were labelled -1. Then we tested against our test data set which had given very good accuracy.

2.4 Results

Here are some accuracy results obtained after training and testing under various variations in parameters:

Gradients	No. of bins	Accuracy(%)
[-1,0,1]	9	86.22
[-1,1]	9	84.86
[1,-8,0,8,-1]	9	87.07
[-1,0,1]	8	86.22
[-1,1]	8	84.86
[1,-8,0,8,-1]	8	86.05

Table 1: Colour Images with L1-norm

Gradients	No. of bins	Accuracy(%)
[-1,0,1]	9	88.1
[-1,1]	9	85.88
[1,-8,0,8,-1]	9	86.73
[-1,0,1]	8	86.05
[-1,1]	8	85.37
[1,-8,0,8,-1]	8	86.56

Table 2: Colour Images with L2-norm

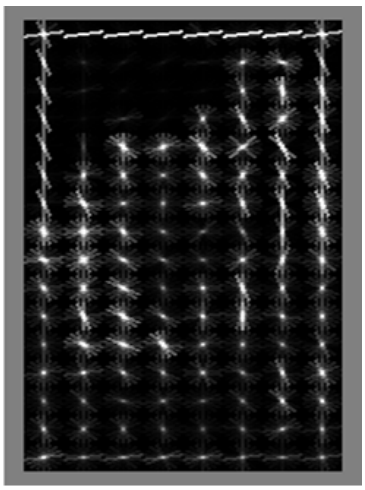
Gradients	No. of bins	Accuracy(%)
[-1,0,1]	9	86.91
[-1,1]	9	87.85
[1,-8,0,8,-1]	9	87.45
[-1,0,1]	8	86.77
[-1,1]	8	87.18
[1,-8,0,8,-1]	8	86.77

Table 3: Grayscale Images with L1-norm

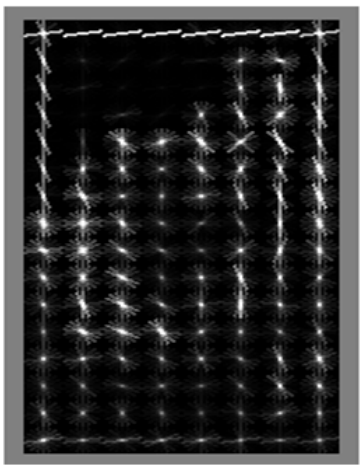
Gradients	No. of bins	Accuracy(%)
[-1,0,1]	9	86.91
[-1,1]	9	87.85
[1,-8,0,8,-1]	9	87.85
[-1,0,1]	8	86.77
[-1,1]	8	87.18
[1,-8,0,8,-1]	8	86.77

Table 4: Grayscale Images with L2-norm

(a) Human in consideration



(b) 9 bins HOG



(c) 8 bins HOG

Figure 1: HOG Desciptors for human

(a) Green box around human in color image

(a) Grren Box around human in Grayscale Image

3 Phase 2

3.1 Introduction

Automatic categorization of human actions in video sequences is very interesting for a variety of applications: detecting activities in video surveillance, indexing video sequences, content based browsing etc. We tried to categorize six classes of human actions, namely running, jogging, walking, boxing, handclapping, handwaving as the time taken to perform the computations for a video is significant. Image structures in video are not restricted to constant velocity and/or constant appearance over time. On the contrary, many interesting events in video are characterized by strong variations of the data in both the spatial and the temporal dimensions. As example, consider scenes with a person entering a room, applauding hand gestures, a car crash. More generally, points with non-constant motion correspond to accelerating local image structures that might correspond to the accelerating objects in the world. Hence, such points might contain important information about the forces that act in the environment and change its structure. In the spatial domain, points with a significant local variation of image intensities have been extensively investigated. Such image points are frequently denoted as "interest points" and are attractive due to their high information contents. we detect interest points in the spatiotemporal domain and illustrate how the resulting spacetime features often correspond to interesting events in video data. To detect spatio-temporal interest points, we used the Harris interest point operators.

3.2 Algorithm

Algorithm 3.1 Space Time Interest Points

Input: A video
Output: Space time Interest Points from the video

1. Convert the video into 3 dimensional matrix along x, y, t
2. Compute L_x, L_y, L_t using any standard differential filter
3. Compute Matrix μ as described above
4. Convolute the obtained μ with a Gaussian window g
5. Calculate the H value of the obtained matrix
6. The local maxima of the obtained H function are taken as STIP's

3.3 Experimental Setup

3.3.1 Dataset

We have used the KTH dataset [5] for training features used for detecting humans actions. We have taken 6 action classes namely bxing, handwaclapping, handwaving, jogging, running & walking. Each action class has 100 videos. These videos have been captured using still camera in homogenous backgrounds at 25 fps rate and are black & white

in colour. The video sequences have spatial resolution of 160×120 pixels and have a length of 4 seconds in average. The videos are performed by 25 subjects in four different scenarios: outdoors $s1$, outdoors with scale variation $s2$, outdoors with different clothes $s3$ and indoors $s4$. All videos are in 'avi' format compressed using xvid codec.

3.3.2 Experiment

First we have obtained the STIP's for each of the 600 videos. These points were stored in a text file which were later used for analysing. We've stored $x, y, t, \sigma_x, \sigma_t, HOG$ features. Next, we created clusters on the obtained HOG features using k-means clustering using various number of clusters. Then we created a histogram from the output of k-means clusters which were then trained using SVM. Though SVM is a two class approach, we've tried to use SVM [6] for multiple classes. We've used one vs. all approach for multi SVM. Build N different binary classifiers. For the i^{th} classifier, let the positive examples be all the points in class i, and let the negative examples be all the points not in class i. Let f_i be the i^{th} classifier. Classify with $f(x) = \arg\max_i f_i(x)$. This classifier resulted in a very good accuracy.

3.4 Results

No. of k means clusters	Action class	Accuracy(%)
50	Jogging	86.5772
	Boxing	81.8792
	Running	87.9195
	Walking	78.5235
	Handwaving	89.2617
	Handclapping	85.2349
100	Jogging	83.2215
	Boxing	79.8658
	Running	81.8792
	Walking	81.8792
	Handwaving	87.9195
	Handclapping	81.2081
400	Jogging	86.5772
	Boxing	80.5369
	Running	84.5638
	Walking	80.5369
	Handwaving	87.9195
	Handclapping	85.2349

Table 5: Accuracy of testing with variation in clusters

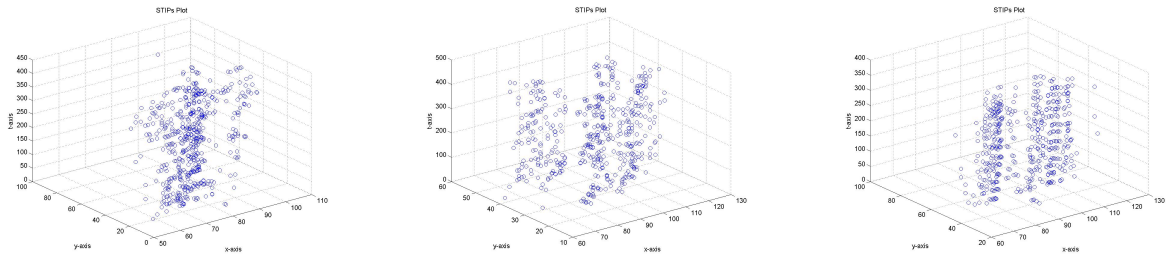


Figure 4: Yellow circle around feature point

Figure 5: Features points along x, y & t plane

Figure 6: Yellow circle around feature point

Figure 7: Features points along x, y & t plane

4 Conclusion & Future Scope

From all the experiments we have conducted to learn human actions, it seems that we have achieved a prety good success in learning(around 80-90%). We could recognize both persons and their actions. We've observed that the algorithm for STIP is a very time taking algorithm and needs to be optimized for faster performance. There were many areas where we've not performed our experiments. The algorithm works only for single person but doesnot accurately work for multiple persons.

References

[1] Ivan Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[2] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *International Conference on Computer Vision & Pattern Recognition*, volume 2, pages 886–893, INRIA Rhône-Alpes, ZIRST-655, av. de l'Europe, Montbonnot-38334, June 2005.

[3] Navneet Dalal. Inria data set of humans and non-humans. http://pascal.inrialpes.fr/data/human/.

[4] Thorsten Joachims. Svm library in c/c++. http://svmlight.joachims.org/. Version: 6.02, Date: 14.08.2008.

[5] Ivan Laptev and Barbara Caputo. Kth data set of action videos. http://www.nada.kth.se/cvap/actions/. Accessed: 2005-01-18.

[6] Chih-Chung Chang and Chih-Jen Lin. A library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm/. Version 3.20 released on November 15, 2014.