Pedestrian And Vehicle Classification

Guided by Dr. Harish Karnick

Group no. 34
Ishita Ankit(13316)
Kanupriya Agarwal(13338)
Shivam Malhotra(13660)
Gaurav Gautam(11917270)

0

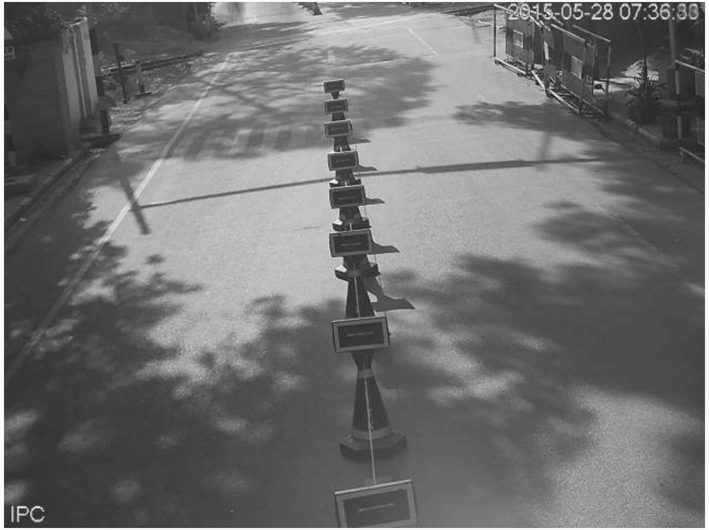
Background subtraction

Algorithms

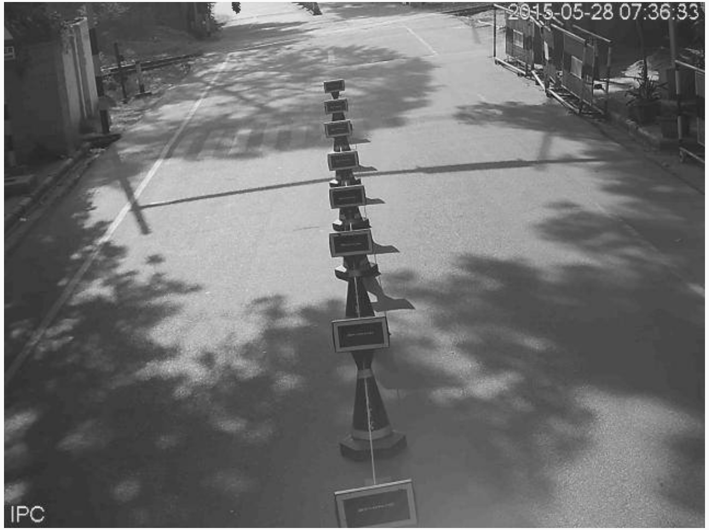
- Average pixel intensity as background
- Mode pixel intensity
- Mixture of Gaussians

Background Images from Various Methods

Average



Mode



MoG (Mixture of Gaussians)

- Then probability of observing the current pixel is given by the following formula:

$$\hat{p}(\vec{x}|\mathcal{X}_T, BG{+}FG) = \sum_{m=1}^{M} \hat{\pi}_m \mathcal{N}(\vec{x}; \hat{\vec{\mu}}_m, \hat{\sigma}_m^2 I)$$

K is the number of distributions,
ω is a weight associated to the ith Gaussian at time t

- Once the parameters initialization is made, the updates are done using the following equations:

$$\hat{\pi}_m \leftarrow \hat{\pi}_m + \alpha(o_m^{(t)} - \hat{\pi}_m)$$
$$\hat{\vec{\mu}}_m \leftarrow \hat{\vec{\mu}}_m + o_m^{(t)}(\alpha/\hat{\pi}_m)\vec{\delta}_m$$
$$\hat{\sigma}_m^2 \leftarrow \hat{\sigma}_m^2 + o_m^{(t)}(\alpha/\hat{\pi}_m)(\vec{\delta}_m^T\vec{\delta}_m - \hat{\sigma}_m^2)$$

- First B Gaussian distribution which exceeds the threshold T is retained for a background distribution. The other distributions are considered to represent a foreground distribution.

$$B = \arg\min_b \left(\sum_{m=1}^{b} \hat{\pi}_m > (1 - c_f) \right)$$

**Improved Adaptive Gaussian Mixture Model for Background Subtraction (Zoran Zivkovic)*

Video: MoG and Average

COMPARISON

AVERAGE

A static method (since average is taken only once), faster than MOG.

The limitation of being static can be somewhat alleviated by updating the average after fixed intervals.

MODE

Intuitively, mode is a better measure for picking background pixels than taking an average intensity, but takes more time and memory.

MOG

Performs better than average and mode but slower.

We did not have a concrete performance measure for background subtraction. Tuning parameters creates little difference visually.

1

Object Detection

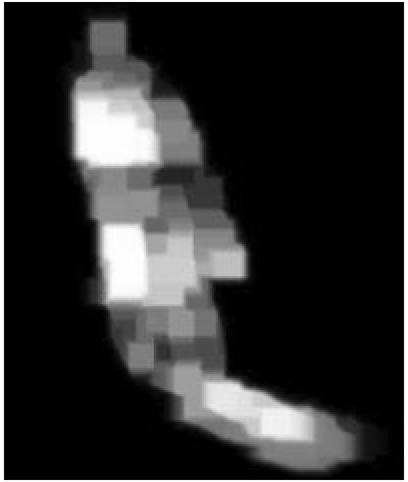
Method

- Applied a combination of morphological effects like erosion and dilation on the background subtracted video
- Obtained contours from the refined foreground obtained from above
- Ignored contours with area less than a threshold
- Constructed bounding boxes using the contours

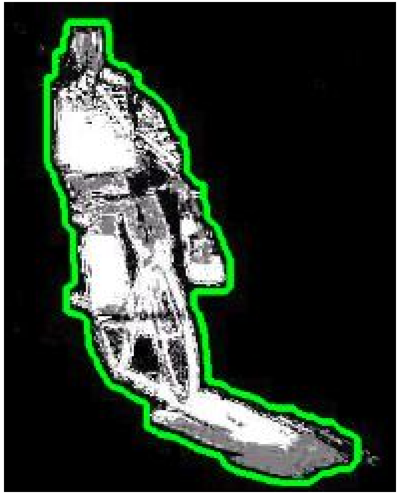
Procedure

Blurred

Morphed



Contours



Comments

- Used Opencv methods to apply morphological transformations
- Implemented a weak form of object tracking for videos, can be improved
- Ordering and parameters of morphological transformations can be finer tuned
- Overlapping objects and shadows not handled

Video:
Box

2

Dataset

Dataset

- Obtained images from the annotated videos
- Cleaned the image set using a custom-written code
- Handpicked images to make a non-repetitive and good quality dataset
- Multiplied the size of dataset by applying transformations on image
- Our dataset consists of 302 persons and 268 non-persons resized to the average size of a person (60 x 166)

3

Classification

Image Representation

- Grayscale pixel values
- Histogram
- Hierarchical Histogram
- HoG (Histogram of Oriented Gradients)
- SIFT 1 (Scale-Invariant Feature Transform)
- SIFT 2

Grayscale Values Feature

Classifier	Accuracy
SVM	80.86 %
Random Forest	78.26 % (max_depth = 16, n_estimators =200)
Adaboost	79.86 % (max_depth = 4, n_estimators =120)

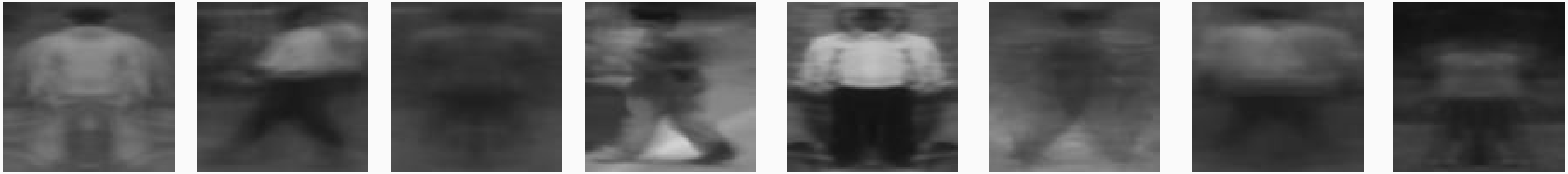
- All images are resized to average size of a person
- Grayscale values of each pixel in an image form the feature vector
- 60 x 166 dimensional vector

K-means Clustering and k-NN

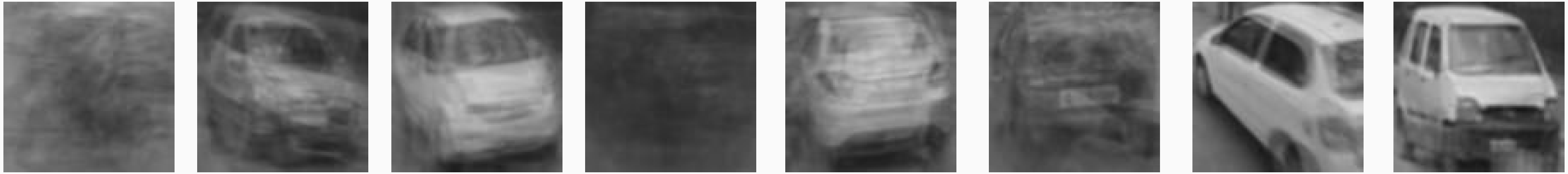
Classifier	Accuracy
k-NN	78.35 % (k = 3, n_means = 8)

- K-means clustering done for both classes (K=8 for each class)
- Cluster centres used to classify images by using k-Nearest Neighbours Algorithm and majority vote

Class Representatives obtained from k-means Clustering



Class 1: Person



Class 2: Non-Person

Histogram Feature

Classifier	Accuracy
SVM	76.11 %
Random Forest	75.98 % (max_depth = 16, n_estimators =200)
Adaboost	79.04 % (max_depth = 4, n_estimators =120)

- Histogram of grayscale intensity values created with 128 bins
- 128 dimensional vector

Hierarchical Histogram

- Multi-level histograms are created for images and appended together
- For n levels (level 0 to n-1) and b bins, feature vector size = $b*\{(4)^n-1\}/3$

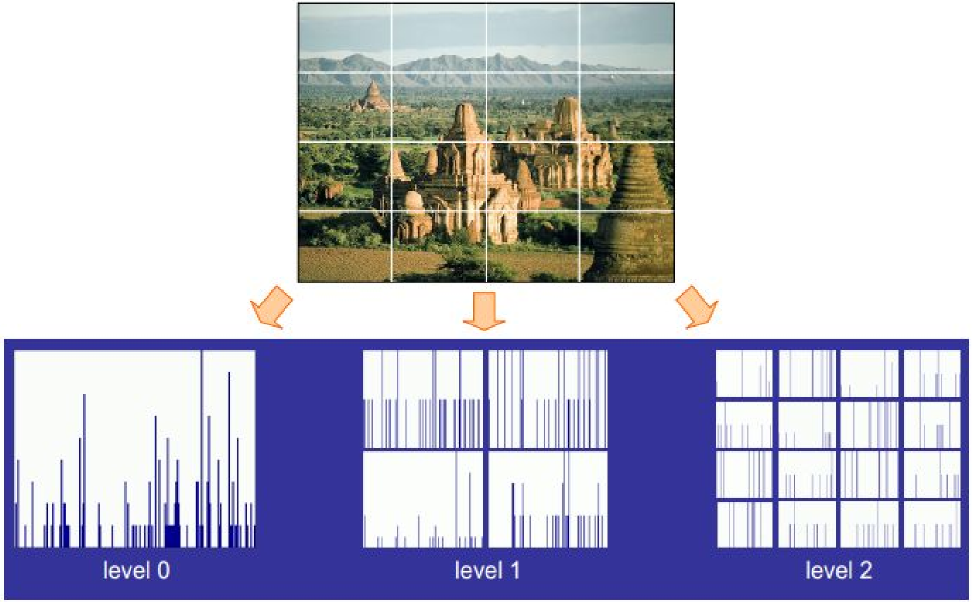


**Image taken from Grauman & Darrell (2005)*

Hierarchical Histogram

Classifier	Accuracy
SVM	78.35 % (num_bins = 128, num_level =3)
Random Forest	79.85 % (max_depth = 16, n_estimators =200) (num_bins = 128, num_level =3)
Adaboost	77.61 % (max_depth = 4, n_estimators =120) (num_bins = 128, num_level =3)

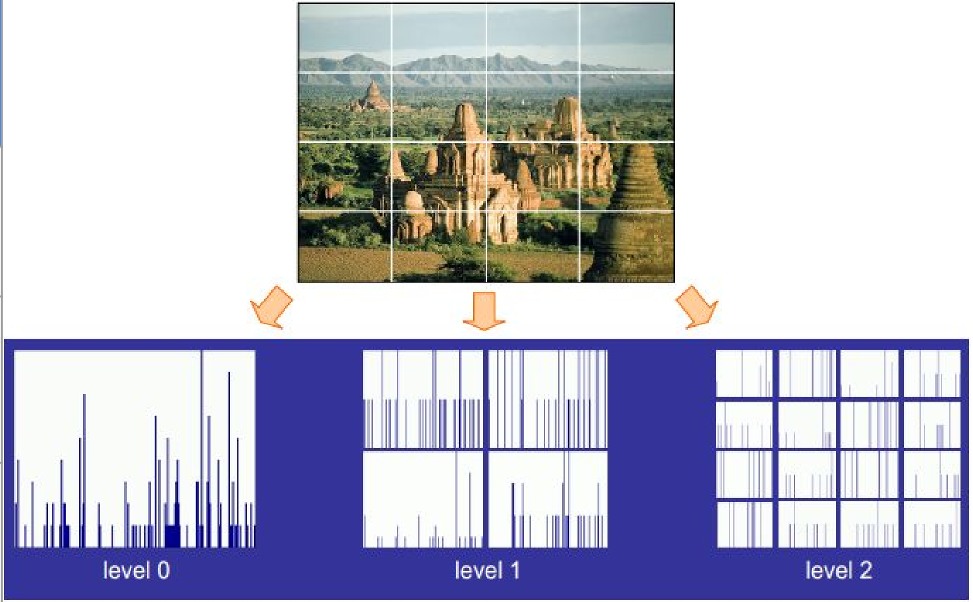
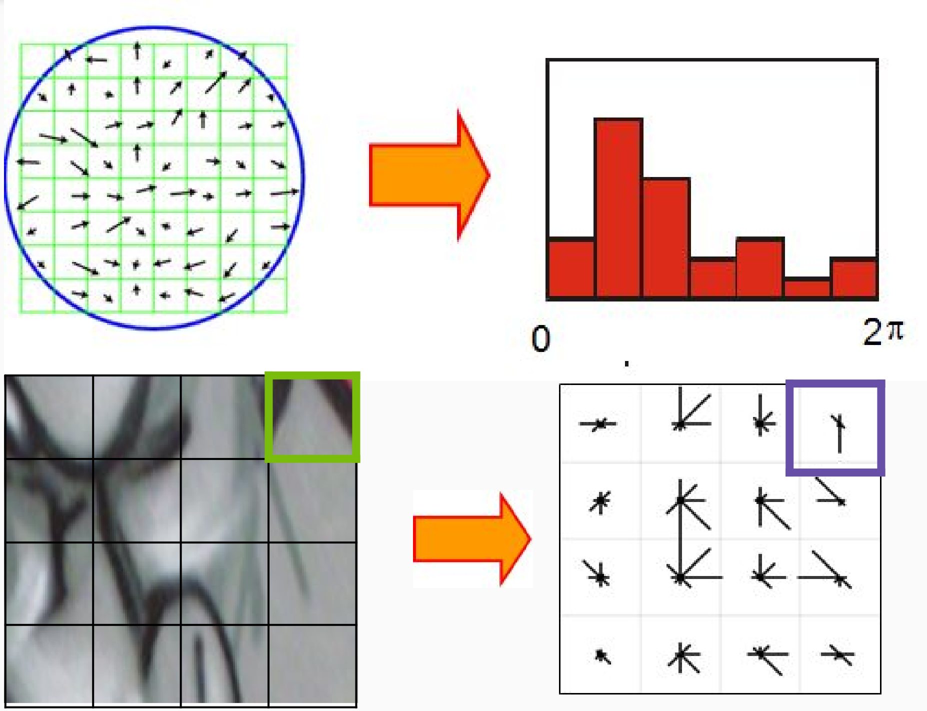


**Image taken from Grauman & Darrell (2005)*

HOG Features

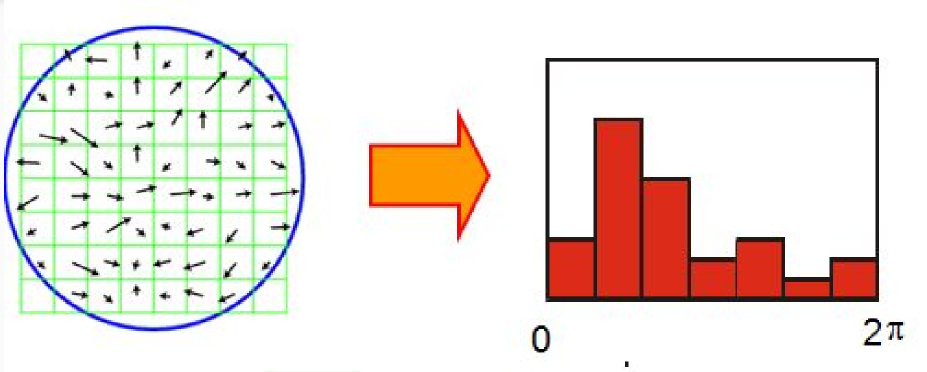
- HoG feature descriptors are extracted from each image with gradient orientations=8, pixels per cell=(16,16) and cells_per_block=(1, 1)

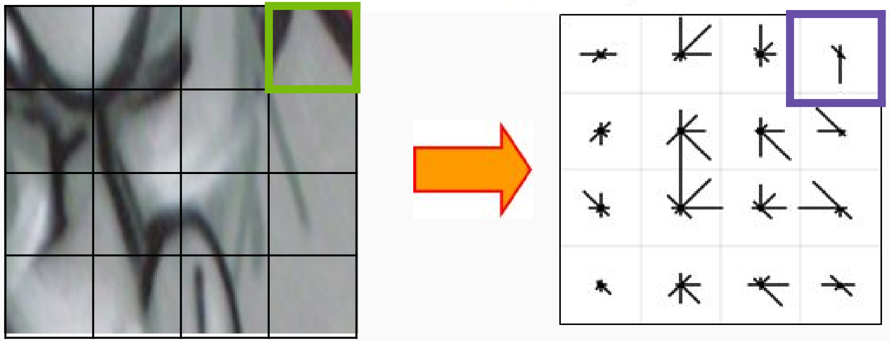


* *images from slides by Deva Ramanan and Kristen Grauman*

HoG Features

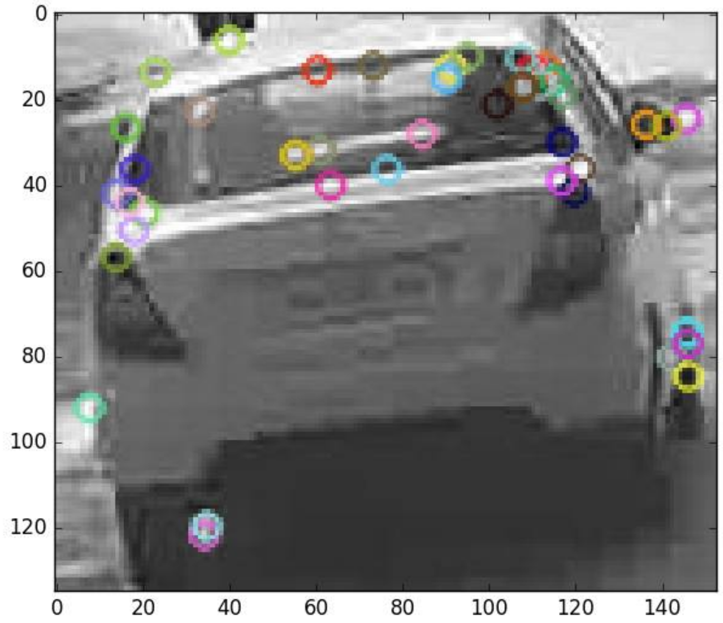
Classifier	Accuracy
SVM	95.37 %
Random Forest	89.60 % (max_depth = 16, n_estimators =200)
Adaboost	91.93 % (max_depth = 4, n_estimators =120)





* *images from slides by Deva Ramanan and Kristen Grauman*

SIFT Interest Points



SIFT 1: Description of Feature

Classifier	Accuracy
SVM	67.76 %
Random Forest	71.07 % (max_depth = 16, n_estimators =200)
Adaboost	73.55 % (max_depth = 4, n_estimators =120)

- Selects 20 SIFT interest points from each image
- Appends all the SIFT features to give one feature per image
- 128 x 20 dimensional descriptor vector
- A classifier is run on these vectors
- For a test image, 128 x 20 vector

Comparing Results from different Image Representations

	Gray Image	Histogram	Hierarchical Histogram	HOG	SIFT 1
Clustering and k-NN	78.35 %	-	-	-	-
SVM	80.86 %	76.11 %	78.35 %	**95.37 %**	67.76 %
Random Forests	78.26 %	75.98 %	79.85 %	**89.60 %**	71.07 %
Adaboost	79.86 %	79.04 %	77.61 %	**91.93 %**	73.55 %

4

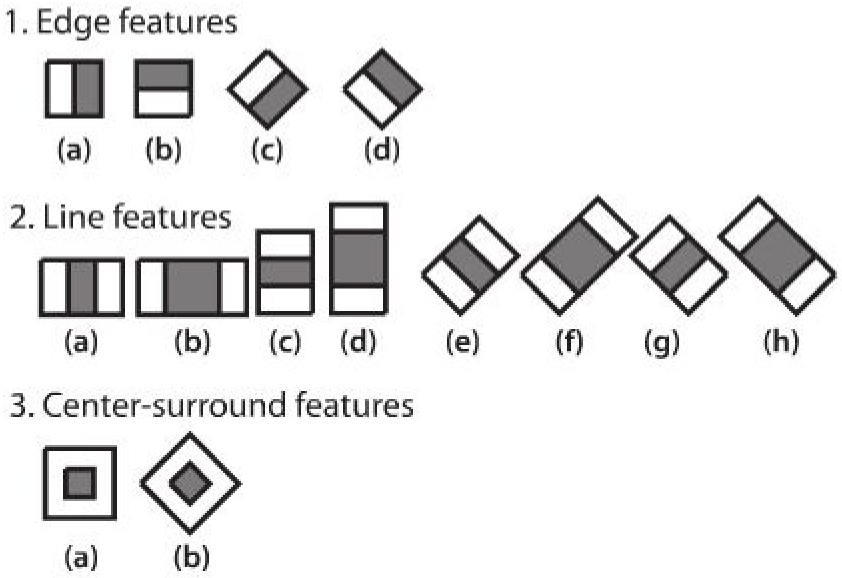
Attempts that didn’t Work

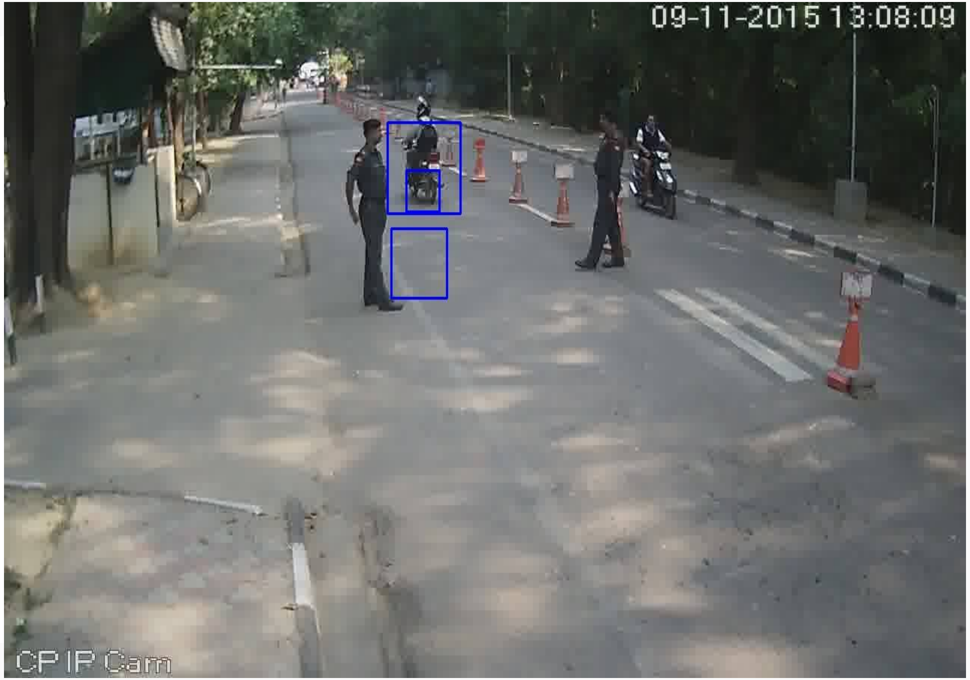
SIFT 2: Description of Feature

Classifier	Cross Validation Accuracy
SVM	40.02 %
Random Forest	% (max_depth = 16, n_estimators =200)
Adaboost	% (max_depth = 4, n_estimators =120)

- Selects a maximum of 10 SIFT features from every image
- Every SIFT feature is given the label of corresponding image
- A classifier is run on these vectors
- For a test image, all SIFT features are classified and class label is given by majority vote

Haar Classifier





Thank You