VIDEO SURVEILLANCE

NANDIT JAIN
NISHANT GUPTA
SARTHAK GARG
TONMOY JYOTI BORAH

VIDEO SURVEILLANCE

- We have implemented a three stage pipeline for this project

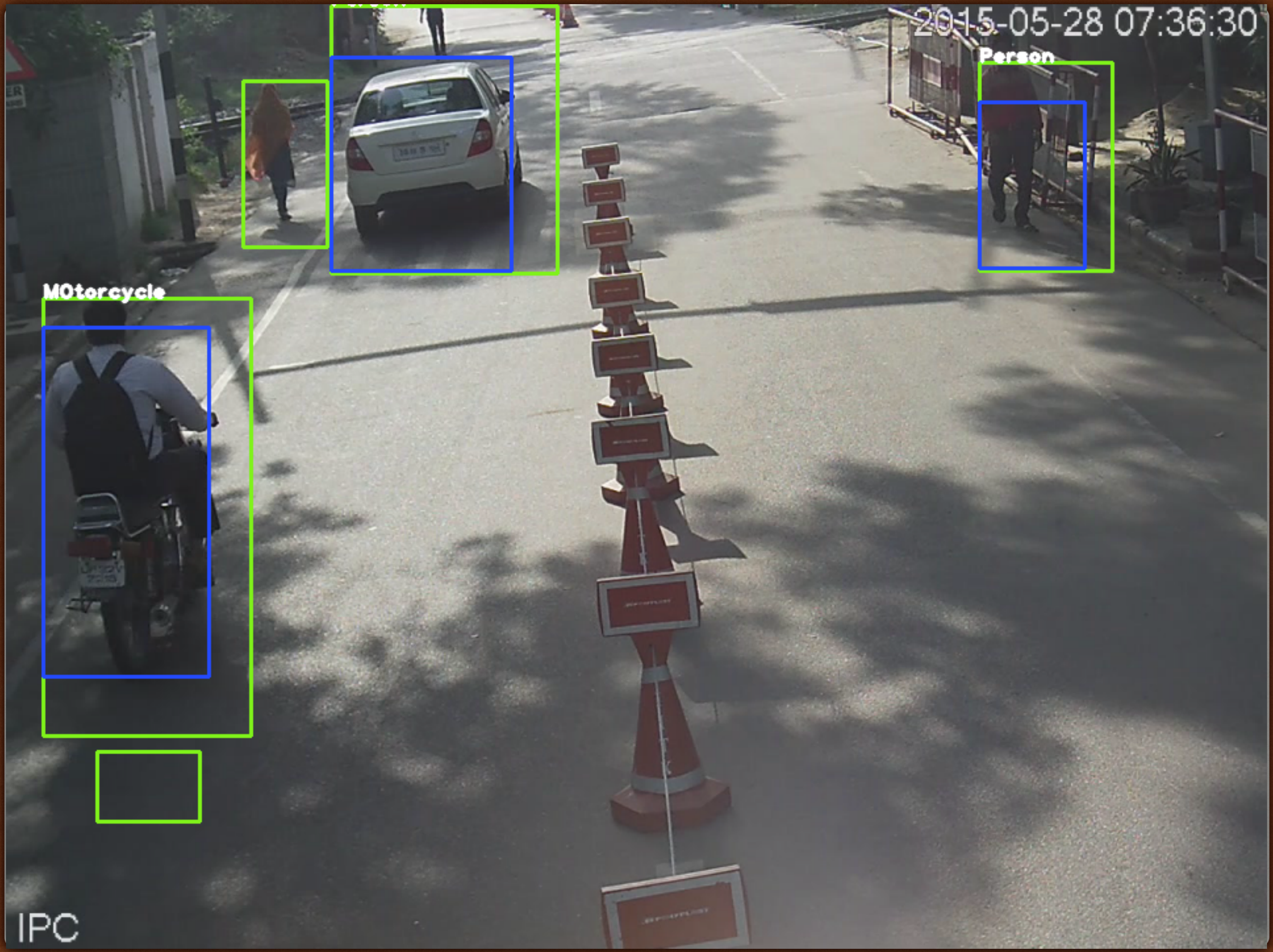
1. Capturing regions of interests (ROI) in video
2. Extracting features from these ROIs
3. Building a classifier and training it on these features

CAPTURING ROI FROM VIDEO

- First to cancel background we have used background subtractor mixture of gaussians.
- BackgroundSubtractorMOG2(history=500,varThreshold = 30, detectShadows = True). We tuned two three parameters for good performance
- We then converted image to gray scale, blurred and resized it and applied Morphological close operation (for seperating partially merged ROI)
- We applied thresholds and calculated contours to generate bounding boxes around the ROI.

CAPTURING ROI FROM VIDEO

- The bounding boxes obtained are a little to wide and contain noise.
- We then applied sliding video along with pyramid of gaussians to capture multiple frames from a single box
- Each captured frame is classified by a classifier (following next), and it’s confidence score is calculated
- This information is sent into the non maximum suppresion algorithm which isolates the bext frames in which object can be present. (suppreses information that is not part of local maximum)


2015-05-28 07:36:30
Person
MOtorcycle
IPC

EXTRACTING FEATURES

- We have tried two techniques: Histogram of oriented Gradients (HoG) and Scale invariant feature transform (SIFT)
- SIFT: Different images give different number of descriptors, which need to be for classification. For this we need to use mini batch k means for clustering the descriptors. The modified feature vector is then simply the frequency count of all the descriptors lying in each cluster.
- HoG: The number of features given by hog = f(no. of pixels). To make this uniform across all images, we resized them into a standard size of 100 x 100

BUILDING A CLASSIFIER

- We have tried several multi-class classifiers.
- SIFT: LinearSVC and RandomForestClassifier, but preliminary tests showed much better accuracy for HoG features
- HoG: LinearSVC, RandomForestClassifier, AdaBoost

TUNING THE PARAMETERS

- HoG features
- hog(orientations = .., pixels per cell = .., cells per block
- The row header is (pixels per cell, cells per block)
- The matrix entry denotes the cross validated accuracy

	[8, 8] and [3,3]	[4,4] and [3,3]	[8, 8] and [2, 2]	[2, 2] and [1, 1]
Linear SVC	0.8886	0.8785	0.8763	0.8651
RFC (200)	0.8231	-	0.8169	-
RFC (250)	0.8213	-	0.8209	-
RFC (275)	0.8307	0.8231	-	-
RFC (300)	0.8237	-	-	-

TUNING THE PARAMETERS

- SIFT features
- Cluster sizes : MiniBatchKMeans(n_clusters)

Cluster Size	Accuracy(Linear SVC)	Accuracy(RFC(275))
100	0.603	-
200	0.5663	0.6267
300	0.5715	0.6475
400	0.5775	0.6419
500	0.5861	0.6522
600	0.5732	0.6410
700	0.5749	0.6371
800	0.56155	0.6358
900	0.6060	0.6341
1000	0.5956	0.6341

FINAL RESULTS

- Best parameters used from previous slides

Parameters Used	Cross Validation	Testing on new video
Linear SVM pixels per cell: [8, 8] cells per block: [3, 3] One vs Rest	0.886	0.5255
RFC(200)	0.8231	0.5802
RFC(275)	0.8307	0.5733
Linear SVM pixels per cell: [8, 8] cells per block: [3, 3] One vs One	-	0.5955

DEEP LEARNING

- We also tried pretrained models based on deep learning
- We used the faster r-cnn approach proposed by Ross Girshik
- The results were much better than in the previous techniques
- All three stages of pipelines were executed with above 95% accuracy
- Some results are attached


person 0.912
person 0.888
person 0.944
car 0.997
person 0.984
motorbike 0.944
motorbike 0.998
person 0.971
CP IP Cam
02-09-2015 09:02:32