Methods and Experiments of Pedestrian Detection

Carnegie Mellon University
The Robotics Institute

Xingchen Yu
Advisor: Christoph Mertz

xingchen9382@gmail.com
B520 NSH

Introduction

The topic of this RISS project is pedestrian detection in images. There are many machine vision approaches for detecting pedestrians. We want to use OpenCV tools to implement, train, and evaluate detectors using Haar and Hog features[1] .

Challenges

- Wide variety of articulated poses
- Variable appearance/clothing
- Complex backgrounds
- Unconstrained illumination
- Occlusions, different scales

Applications

- Pedestrian detection for smart cars
- Film & media analysis
- Visual surveillance

Database

Database Name	INRIA	CVC02	CVC07
Positive samples	3506	2054	4824
Negative samples	From Internet(29552)		

Positive samples 40*98

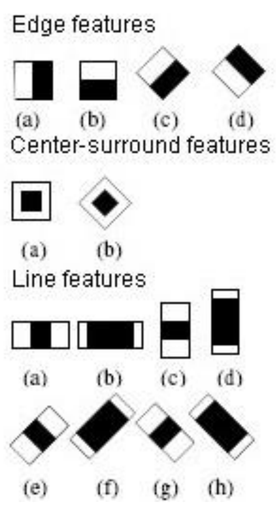
Negative samples

Methods

1. Collect enough samples to train classifiers.
2. Input classifier to the detection program and analyze images.
3. Compare detection results with groundtruth, evaluate experiments using miss rate and false positive per image.

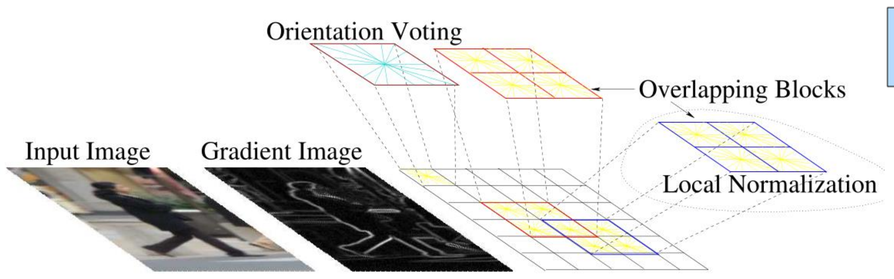
① HAAR-like Feature

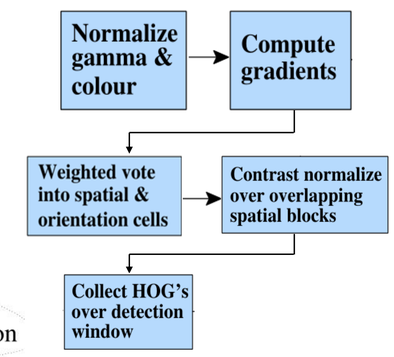
HAAR-like Feature are the difference of the sum of pixels of areas inside the rectangle, which can be at any position and scale within the original image.



② HOG Feature

HOG feature means local object appearance and shape within an image can be described by the distribution of intensity gradients or edge directions[2] .





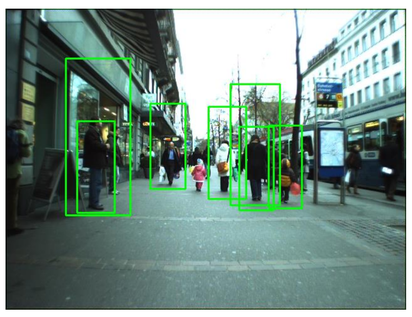
Experiments

HAAR+Adaboost

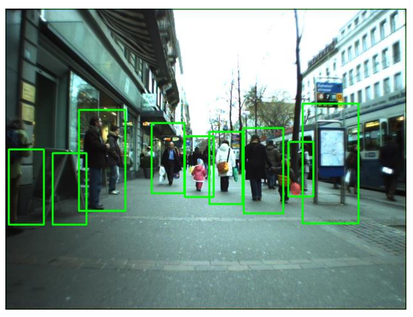
We tested the dependency of the classifiers on the number of training samples and the number of stages. We did four experiments and got their results. And we used miss rate and false positive per image to express the detection results. We also compared our results with the results in the paper[1]. In the paper, they evaluated different methods to detect pedestrians. The average analysis time per image is 300ms.

Test	Database	Total Pos Num	Pos Num per stage	Total Neg Num	Neg Num per stage	Stage
1	INRIA&CVC02	1000	800	8089	2400	20
2	INRIA&CVC02	5560	4000	16552	12000	20
3	INRIA&CVC07	8330	7000	29552	21000	20
4	INRIA&CVC07	8330	7000	29552	21000	25

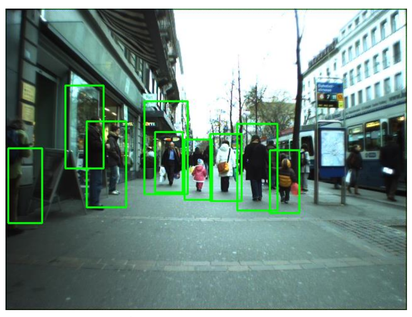
Test	Curve Color	Training Time
1	Red	12 hours
2	Green	42 hours
3	Blue	73 hours
4	Orange	95 hours



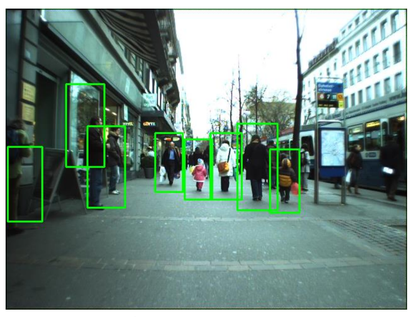
1st test example



2nd test example



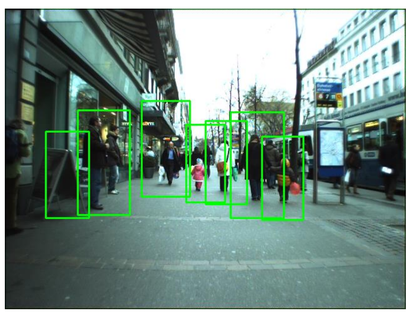
3rd test example



4th test example

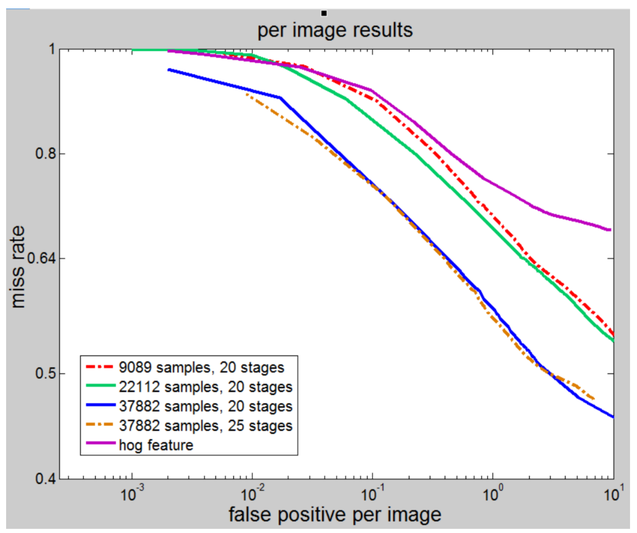
HOG+Adaboost

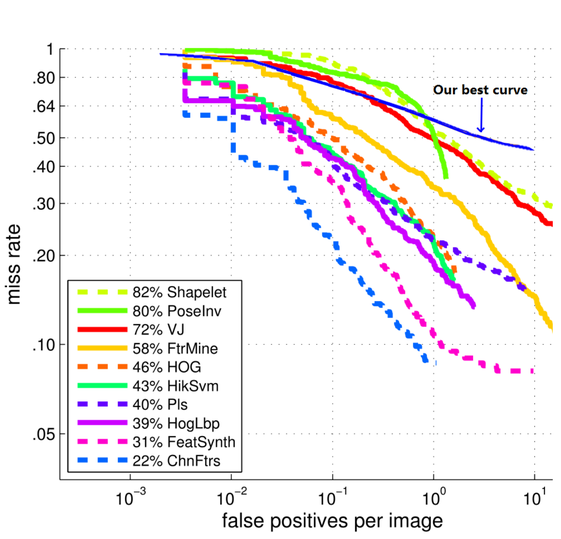
Use pre-trained classifier called hogcascade _pedestrians.xml in OpenCV



Detection Results Curves

The curves in the left graph show results of our five experiments, the curves in the right graph show the comparison of our best result and the results in the paper[1]. The results in the paper are better than our experimental results, we also need to consider some other effects, such as the difference of test samples.





Conclusion

After study, we got some results in computer vision and pedestrian detection. During this process, we did lots of experiments and trained couple of classifiers. More samples, better results. Actually, the accuracy of experiments still has enough space for further improvement.

References

[1] Dollar P, Wojek C, Schiele B, et al. Pedestrian Detection: An Evaluation Of The State Of The Art[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, 2012, 34(4):743-761.
[2] Dalal N, Triggs B. Histograms of oriented gradients for human detection[J]. Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, 1(12):886-893.