Linear combinations of simple classifiers for the PASCAL challenge

Nik A. Melchior and David Lee
16–721 Advanced Perception
The Robotics Institute
Carnegie Mellon University
Email: melchior@cmu.edu, dlee1@andrew.cmu.edu

***Abstract*— Our class project for the 16–721 Advanced Perception was an entry in the PASCAL Visual Object Classes Challenge 2006. The goal of this challenge is to determine whether an object from one of ten classes appears in a given image. Labelled training data was provided, and participants were free to use any method. We chose to implement a bag–of–words classifier using color, shape, and texture information. Adaboost was used to determine the best weights for combining classifiers based on these simple descriptors. This report details the development of the overall classifier and discusses results of the challenge.**

I. Introduction

This paper describes our entry in the PASCAL Visual Object Classes Challenge[1]. This challenge was divided into two different competitions, which were termed classification and detection. The goal of the classification challenge, in which we participated, was to determine the confidence that a given object appears in an image. Participants were free to choose a subset of the ten object classes selected for the challenge. Submissions for the classification challenge consisted a real–valued confidence that an object of the given class appears in each test image. This permits the calculation of Receiver Operating Characteristic (ROC) curves and the area under curve (AUC), which is used for quantitative comparisons between entries. The detection competition required participants to determine the bounding box of each detected object in the image.

The competition began on February 14 with the release of development code and 2618 labelled training images. For each training image, the bounding box and pose of each instance of an object class was included. The object classes are bicycle, bus, car, cat, cow, dog, horse, motorbike, person, and sheep. The test set consisted of an additional 2686 images released on March 31 without ground truth labels. Unlike typical computer vision data sets, the images used in this challenge were rarely well–composed photographs of the objects to be detected. Objects often appear occluded, partially out of frame, silhouetted, or very distant.

We began the challenge hoping to develop an entry based on a relatively simple classifier. The difficulties of this particular data set led us to believe that no single feature, or set of features, would be sufficient to classify every image. Image features based on color, texture, and shape could each contribute to the overall classification, but some features would be more helpful in some classes than in others. For example, most of the images of sheep and cows appear in the context of green grass and blue skies, so color information alone should help classify these image. The images in figure 1 illustrate some of the difficulties mentioned above.

We would like our classifier to recognize and benefit from the ability to classify in the lower–dimensional space of color features without confounding by texture and shape. To this end, we chose to perform classification independently with each of the seven possible combinations of our simple features. The individual classifiers used a bag–of–words [1] approach, and AdaBoost [2] was used to discover the appropriate weighting of these classifiers for each object class. The single–feature classifiers were based on simple histograms of colors, the textons developed by Martin [3], and the shape information obtained from Histograms of Oriented Gradients (HoG) [4].

Each of these features, and the overall classifier are described in depth in the following section. In section III, we present the results of the classification challenge, including the comparison of our classifier with other entries. Finally, section IV describes possible improvements to our strategy.

II. Algorithm

We classify patches of multiple scales within an image independently (so-called bag-of-words approach). Our classifier is a combination of fourteen weak classifiers, seven classifiers trained on the bounding box of the object of interest and seven trained on the entire image including the background. A weighted sum of these fourteen weak classifiers, where the weights are chosen by AdaBoost, gives us a confidence for each patch. Confidence of all the patches in an image is averaged to give a final confidence of the image

A. Bag–of–Words Classifier

The bag–of–words classifier extracts features from image patches (or sometimes uses the actual pixel values of the patches themselves) for classification. The hope is that the patches are more provide more information for classification than the more traditional interest point features, without losing generalization ability.

[1] http://www.pascal-network.org/challenges/VOC/voc2006/index.html

(a) Object in silhouette

(b) Color and texture are misleading

Fig. 1. Some difficult images

The features which we extract from patches are described below. The patches themselves are 16 by 16 pixel squares extracted from the image at three different scales. We evaluated the algorithm when features were extracted from the entire image as well as limiting the features to the ground truth bounding box of the object. For the sake of speed, the image patches used for training were nonoverlapping. During testing, the improvement of a sliding window approach did not justify the added running time, but we added an additional rescaling of the image.

Next, the features extracted during training must be stored for comparison with those of the test images. Rather than store all extracted features, the typical approach is to cluster the features and store a limited number of cluster centers. For each of our fourteen simple classifiers, we created positive and negative dictionaries from these cluster centers. Negative dictionaries were built using features extracted from entire images which did not contain the object class. To accommodate AdaBoost's requirement to weight training examples at each iteration of training, we implemented an extension to kmeans which altered the distances to examples based on their weights.

B. Descriptors

Three types of descriptors are computed on 16-by-16 pixel image patches to get a reduced dimension descriptors.

1) Color: Histogram in RGB space is used as the color descriptor. We choose the top 2 entries in the histogram to get a 6 dimensional descriptor for color.

2) Texture: Histogram of textons are used as the texture descriptor. Textons are trained with a small set of images. A set of filters in a filterbank is applied to images and the response of filters are clustered into groups to get 32 textons. We used textons that were already trained by Martin et al. [3] When an image is given, each pixel in the image is assigned to be one of the 32 textons, by applying the same filterbank and finding its closest texton. Histogram of textons gives us a 32 dimensional texture descriptor.

3) Histogram of Oriented Gradients: We have used the method proposed by Dalal et al. [4] with some minor difference in parameters. 16-by-16 pixels image patch is divided into four subregion of 8-by-8 pixels. For each subregion, gradient angle is binned into 18 bins (360 degrees/20 deg/bin) to get a histogram. Histograms from four regions are concatenated to get a 72 dimensional descriptor.

C. Combining Classifiers

The results of the individual bag–of–words classifiers are combined using AdaBoost. We used the Matlab implementation provided by the MSU Graphics & Media Lab, Computer Vision Group, `http://graphics.cs.msu.ru`. We used a variant of AdaBoost included in this toolbox called Modest AdaBoost [5], which uses regularized tradeoff to help avoid overfitting and improve generalization.

We experimented with several techniques for combining the results of the fourteen simple classifiers, but the best results were obtained by simply giving one vote to each feature with each classifier. We expected to see better results by weighing the votes based on the classification margin or using a nearest neighbor ratio test to determine which features were truly discriminative, but no variation of these tests improved classification results.

III. Results

Twenty teams submitted entries to the classification challenge, and most teams tackled all ten object classes. Although our results were the best among the few entries from Carnegie Mellon, our algorithm performed near the middle of the pack in the final results. An example of the final ROC curves of all entries is shown in figure 2 for the sheep class. Our entry is labelled AP06_Lee. Table II summarizes our performance compared to the winning entry in each class. We were able to compute the results shown for the validation set prior to submission using the ground truth labels provided with the training data. The test results were computed by the competition organizers, and the ground truth for this image

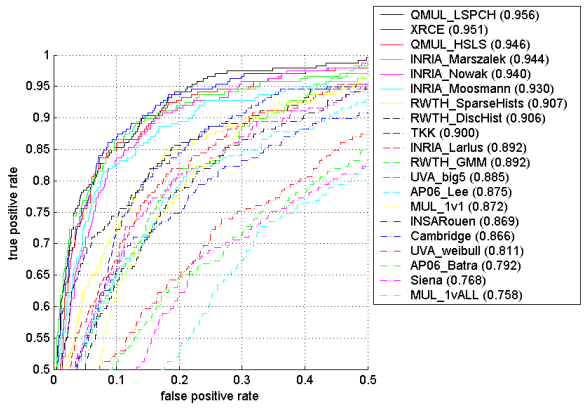


Fig. 2. A collection of ROC curves for one object class

set has not yet been released. The results of all entries are temporally available at `http://www.robots.ox.ac.uk/~me/voc2006res/`. A full report describing the results and each participant's approach is expected to be available shortly from the challenge website.

Figure 3(a) shows 100 images from the validation set that scored the highest for class sheep, with bottom right being the most confidant and top left the least confidant. Figure 3(b) shows 100 images with the lowest score, again with top left being the least confidant. There were a total of 132 images out of 1341 images which contained sheep in the validation set.

Table I shows the weights assigned to fourteen weak classifiers by AdaBoost. Higher number means the feature is more discriminative, and zero means the feature is of no use. Negative number means that it is worse than random guess. Color+HOG and color+HOG+texture are usually the best feature and HOGs in general are good. Color by itself is not very useful but becomes useful when combined with other features. Classifiers trained on the bounding box of the object is generally better than classifiers trained on the entire image including the background, but still background provides some help.

IV. Conclusions

We have developed a simple but effective classifier for the PASCAL challenge. We have won the CMU mini competition and our results were better than the winner of 2005 PASCAL Challenge. We were near the middle among the participants in the 2006 challenge. We think that our success was due to noticing that no single feature can be very effective and thus using multiple features, color, texture, and histogram of oriented gradients. Our contribution is also in developing a way to combine multiple features.

Our bag-of-words approach, however, is possibly losing valuable information by not considering any spatial relation between patches. We have tried to get a rough estimation of the object center by having each patch vote in image space for the candidate object center and utilize that information by weighting votes of patches that estimates the object center correctly. We have been somewhat successful in estimating the object center 4 but failed to utilize that information to improve the classification rate due to the lack of time.

This work still has potential for improvements, such as better engineered descriptors, more powerful weak classifiers, and using spatial information by estimating the location of object through spatial voting. If we continue on this track, we may get better standing in the next year's challenge.

References

[1] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual categorization with bags of keypoints," in *ECCV Workshop on Statistical Learning in Computer Vision*, 2004, pp. 1 – 22.

[2] R. Schapire and Y. Singer, "Improved boosting algorithms using confidence–rated predictions," *Machine Learning*, vol. 37, no. 3, pp. 297 – 336, December 1999.

[3] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, July 2001, pp. 416–423.

[4] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[5] A. Vezhnevets and V. Vezhnevets, "Modest adaboost — teaching adaboost to generalize better," in *Graphicon*, 2005.

(a) 100 images with highest score

(b) 100 images with lowest score

Fig. 3. Results

Entry		Color	HOG	Texture	Color+HOG	Color+Texture	HOG+Texture	Color+HOG+Texture
Bicycle	obj	0.053	0.358	0.135	0.350	0.180	0.359	0.360
	bg	-0.089	0.218	0.070	0.256	0.112	0.237	0.246
Bus	obj	-0.013	0.493	0.224	0.609	0.373	0.496	0.618
	bg	-0.081	0.368	0.175	0.467	0.243	0.377	0.437
Car	obj	0.168	0.427	0.256	0.491	0.332	0.450	0.500
	bg	0.085	0.223	0.116	0.255	0.213	0.229	0.283
Cat	obj	-0.005	0.369	0.238	0.400	0.252	0.370	0.392
	bg	-0.021	0.257	0.182	0.333	0.249	0.262	0.311
Cow	obj	-0.050	0.385	0.230	0.425	0.335	0.388	0.468
	bg	-0.074	0.218	0.055	0.285	0.213	0.213	0.342
Dog	obj	-0.054	0.344	0.220	0.326	0.190	0.357	0.331
	bg	-0.046	0.219	0.142	0.267	0.113	0.227	0.257
Horse	obj	-0.041	0.334	0.184	0.320	0.205	0.328	0.309
	bg	-0.101	0.167	0.073	0.193	0.074	0.175	0.164
Motorbike	obj	-0.070	0.346	0.161	0.368	0.272	0.356	0.373
	bg	-0.103	0.193	0.095	0.208	0.164	0.201	0.231
Person	obj	0.108	0.283	0.174	0.294	0.191	0.296	0.313
	bg	0.005	0.071	0.034	0.089	0.061	0.074	0.115
Sheep	obj	-0.002	0.362	0.227	0.377	0.294	0.363	0.373
	bg	-0.008	0.215	0.070	0.330	0.293	0.221	0.323

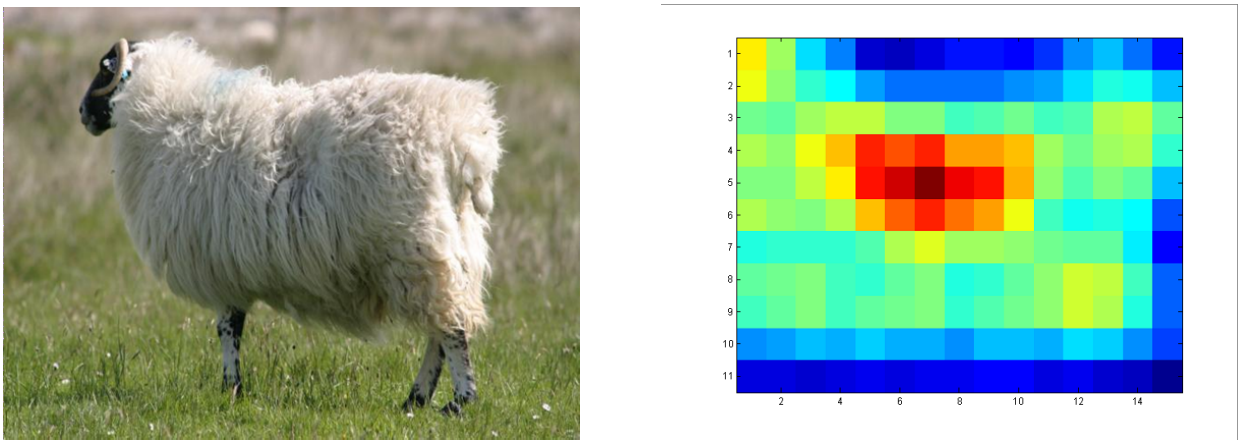
TABLE I

WEIGHTS ASSIGNED TO EACH WEAK CLASSIFIER BY ADABOOST

Entry	Bicycle	Bus	Car	Cat	Cow	Dog	Horse	Motorbike	Person	Sheep
Us (Validation)	0.853	0.900	0.905	0.846	0.873	0.770	0.750	0.802	0.656	0.848
Us (Test)	0.845	0.916	0.897	0.859	0.838	0.766	0.694	0.829	0.622	0.875
Best Test	0.948	0.984	0.977	0.937	0.940	0.876	0.926	0.969	0.863	0.956

TABLE II

CLASSIFICATION ACCURACY AS AUC VALUES



(a) Input image

(b) Estimated Object Center

Fig. 4. Spatial Voting