Computer Vision 1
Final Project: Bag-of-Words based Image Classification

Riaan Zoetmulder
MSc Artificial Intelligence
University of Amsterdam
riaanzoetmulder@gmail.com

Minh Ngo
MSc Artificial Intelligence
University of Amsterdam
nlminhtl@gmail.com

1 Introduction

The purpose of this assignment is to create an image classifier that can classify four different objects; airplanes, motorbikes, cars and faces. Firstly, we will discuss how we constructed our image classification pipeline. We will discuss SIFT descriptor extraction from the various color spaces, constructing a visual vocabulary, turning them into histograms and the classification process. Secondly, we will discuss the experiments we ran. Thirdly, we will discuss the results. Fourthly, we will conclude and discuss further improvements on our experiment.

2 Image classification pipeline

2.1 Data

For experimenting purpose the whole training set of 1864 images that consist of 500 images with airplanes, 465 images with cars, 400 images with faces and 500 images with motorbikes. We used a separate testing set of 200 images (50 images of each class) to evaluate our object recognition pipeline. Training data has been shuffled in a random order.

2.2 SIFT descriptors extraction

In our pipeline, we firstly load the images and check which color space we have to convert them to. For this experiment we used the following color spaces:

- Grayscale
- RGB
- Normalized RGB
- Opponent Colors
- HSV (as a **bonus** part)

For Keypoint-SIFT and Dense-SIFT experiments in gray scale images, standard SIFT and Dense-SIFT descriptors (Lowe, 2004) have been used. To use Dense-SIFT descriptor in images with multiple color layers we applied it for each of the layers and concatenated the features together to form a feature vector of size $D \times 128$ where D is an image dimension. For Keypoint-SIFT an image is firstly converted to gray scale after which key points are computed. For each key point the SIFT descriptor is computed for each layer and combined together as it was done for Dense-SIFT (van de Sande et al., 2010).

2.3 Building a visual vocabulary

SIFT features are already normalized, but after stacking color space descriptors together they become no longer normalized. Therefore after computing them we normalized by calculating the norm of each descriptor and dividing each element by it. We were than capable of running K-means (Mohamad and Usman, 2013) to obtain the centroids.

Subsequently we created a histogram for each picture accordingly, by assigning the features to the centroid it was located closest to. We then ensured that the sum of the histogram entries was equal to 1.

2.4 Training classifiers

Once we had the histograms we standardized them by subtracting the mean of the features and dividing them by the standard deviation of the training data (Hsu et al., 2003). Next we proceeded with training the Support Vector Machines (SVM's) classifiers. For multinomial classification we used 4 one-vs-rest classifiers. We used the full training data for each of the classifiers and transformed the initial labels to binary label vectors depending on the positive class of the particular classifier. Each of the classifiers yielded a confidence level of its decision, therefore we can obtain the final decision by computing the *argmax* from those decisions (Bishop, 2006).

2.5 Classification

After training the SVM's we continued with evaluation of the whole pipeline by classifying each image from the test set and calculating the Precision, Recall, Accuracy, Average Precision and Mean Average Precision. Before we classified the images we ensured that the features were normalized and standardized the same way as the features were before the training of the SVM's. We then classified the images using the histograms.

3 Experiments

Five experiments have been performed. In the first experiment we run Keypoint-SIFT and Dense-SIFT for different color spaces (RGB, rgb, opponent, gray scale and HSV). For Dense-SIFT descriptors we used a step size of 12 and a bin size of 3. 400 randomly picked images have been used for building a visual vocabulary of 400 words. We used SVM classifier with RBF kernel and a slackness variable equal to 3 for all configurations.

In the second experiment the influence of the vocabulary size was considered. We performed classification on the gray scale color space with a varying number of clusters (400, 800, 1600, 2000, 4000). We used the same configuration as the first experiment.

In the third experiment we researched the influence of Dense-SIFT parameters (step size and bin size) in the object classification task. A step size value has been chosen equal to 12, 8 or 6 and a bin size equal to 4 or 3.

In the fourth experiment, we measured the impact of the amount of images used for building the visual vocabulary. A sample size of 400, 600 and 800 images has been used as the input for the K-Means clustering. Experiments were performed using Keypoint-SIFT and an SVM with the configuration described before.

In the last experiment we conducted trials with different kernel types for the SVM classifier namely linear, polynomial (degree of 3), tanh and RBF on the gray scale color space.

4 Results

4.1 Varying Color spaces

Mean average precision obtained for classification pipelines with different color descriptors are reported in the figure 1. There is a significant performance improvement of Dense-SIFT descriptors over SIFT descriptors on keypoints. It can be explained by the fact that if descriptors on positions differ from key points, dense sampling will capture information that distinguishes one object from another. We did notice improvements when using color spaces that are invariant to lighting conditions for Keypoints-SIFT, but with the use of a Dense-SIFT descriptor this advantage becomes insignificant. In fact, we have noticed a small improvement of accuracy when applying SIFT on HSV images instead of RGB (97% against 95%), but on the other hand the latter beats the former in term of mean average precision

(90.1% against 91.8%). In spite of the fact that normalized RGB color space is invariant from shading and shadow, we didn't notice significant differences between RGB and rgb. We believe that our test data is trivial and is not affected so much by these problems.

Average precision separately for each class is reported in the figure 2. SIFT features on the HSV color space perform the best for the motorbike, face and car classes, the rgb color space outperform others in case of the airplane class. Consequently, one solution to improve performance is to use rgb color space for the airplane detection and HSV for other classes and then combine those 4 to reach the final decision.

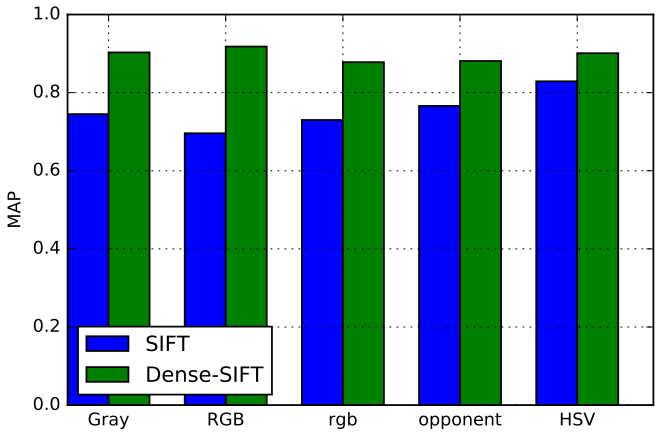

Figure 1: Mean average precision for the classification pipeline with SIFT and Dense-SIFT descriptors for different color spaces.

4.2 Vocabulary Size

In this experiment we varied the number of centroids (vocabulary size) that the K-means algorithm used. We used gray-scale images, with a sample size of 400 and SIFT-keypoints. The results are shown in figure 3. As can be seen using a vocabulary size of 400 is the best option. Increasing vocabulary size only adds noise and decreases MAP.

4.3 Key Points vs Dense sampling

We wanted to know which type of sampling gave better performance; Keypoints or dense sampling. In this experiment we kept the number of centroids and sample (number of images) fixed, we also used gray scale only. For the dense sampling we only varied the bin size and step size. Results are shown in figure 4. As can be seen dense sampling consistently outperforms keypoint sampling.

4.4 Sample size

We also wanted to see what the impact was of using different sample sizes on the mean average precision. We therefore varied the sample size, whilst keeping the rest of the variables constant. The results are shown in figure 5. As can be seen using a sample of 400 images is optimal. A possible explanation is that using more images, creates more features that can be explained by a different amount of latent variables. As such a different amount of centroids must be used on different sample sizes.

4.5 Varying SVM Kernels

We had four different types of kernels with which we could run the SVM classification; a linear kernel, a polynomial kernel, an rbf kernel and a tanh kernel. We kept all variables constant as in the previous experiments and varied only

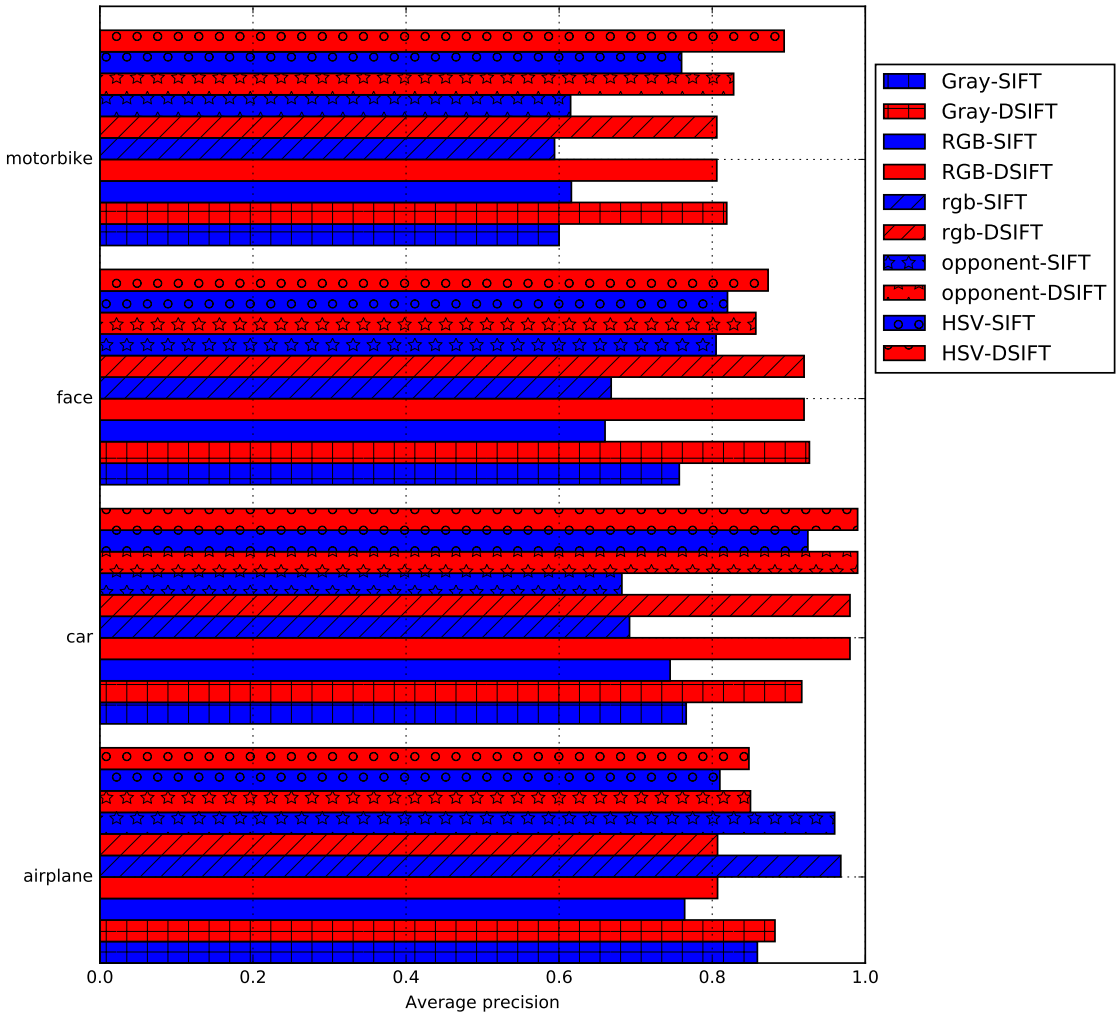

Figure 2: Descriptor performance split out per category.

the kernels with which we predicted. Results are shown in figure 6. As can be seen RBF has the highest MAP. This is not surprising since our problem is of a non-linear nature.

5 Conclusion

From the preceding experiments several conclusions can be drawn. Firstly, we have found that RGB in average performs best for this image dataset. Nevertheless, some class of our data seems to work better in the HSV color space and some works better in the rgb color space. A classifier built with combination of those can obtain better performance. Secondly, we have found that using a vocabulary size of 400 works better than using larger vocabulary sizes, because that makes visual words less discriminative. Thirdly, when it comes to the performance of keypoints and dense sampling, dense sampling consistently outperforms keypoints but is much more computationally expensive. Fourthly, we have found that a sample size of 400 is sufficient to obtain good results, larger sample sizes decrease performance and is time and resource consuming. Fifthly, because of the non-linear nature of our problem, an rbf kernel works best for classification.

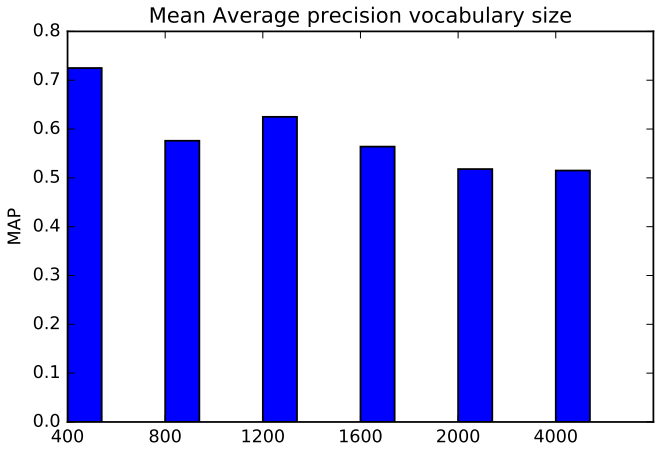

Figure 3: MAP for the centroid number of 400, 800, 1200, 1600, 2000, 4000.

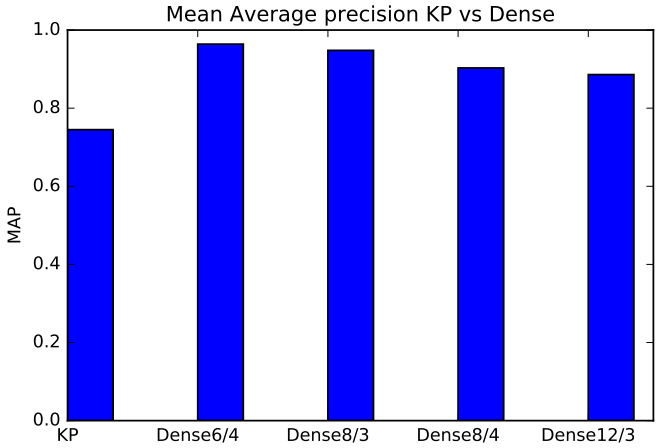

Figure 4: MAP for the centroid number Keypoints and Dense sampling with varying bin and step sizes.

6 Future works

Despite these promising results there remains considerable room for improvement. The first one of these improvements, would be to combine color spaces together. Given that RGB and HSV combined with dense sampling yield good results on the test sets separately. It could be possible to combine the Hue part of HSV with RGB so that they RGB has less of a problem with highlights for example. The second improvement would be to use spatial pyramids, because they can erode the influence of features extracted from the objects background. The last improvement we propose is to replace K-Means clustering by mixture models to avoid the hard cluster assignment.

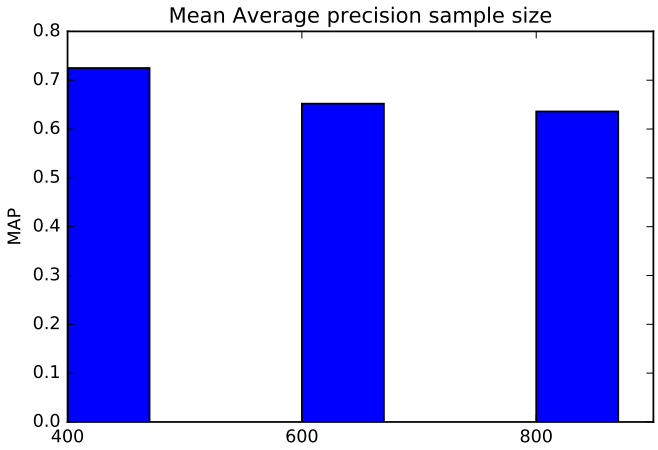

Figure 5: MAP for varying sample sizes, using gray scale images, 400 centroids and key points.

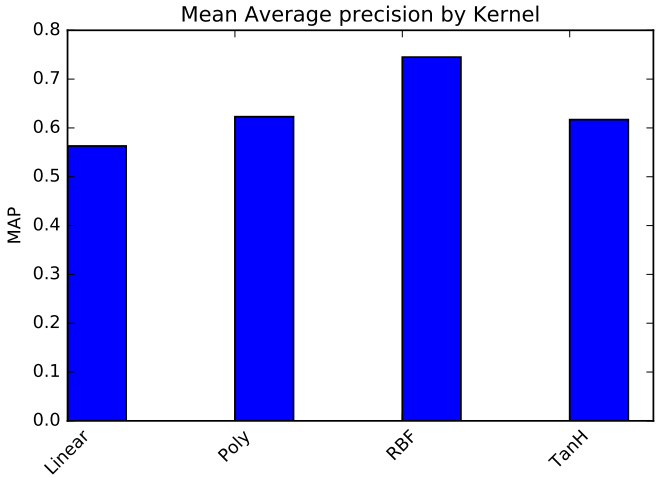

Figure 6: MAP for varying sample sizes, using gray scale images, 400 centroids and key points.

References

C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

C. W. Hsu, C. C. Chang, and C. J. Lin. *A practical guide to support vector classification*. Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 2003. URL http://www.csie.ntu.edu.tw/{~}cjlin/papers/guide/guide.pdf.

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004. ISSN 0920-5691. doi: 10.1023/B:VISI.0000029664.99615.94. URL http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94.

I. B. Mohamad and D. Usman. Standardization and its effects on k-means clustering algorithm, Aug 2013.

K. van de Sande, T. Gevers, and C. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(9):1582–1596, Sept. 2010. ISSN 0162-8828. doi: 10.1109/TPAMI.2009.154. URL http://dx.doi.org/10.1109/TPAMI.2009.154.