COMPUTER VISION AND PATTERN RECOGNITION COURSEWORK REPORT 2023/2024

Abstract

This is the report of the computer vision and pattern recognition coursework September 2023. The methodology and thought process of each exercise are described in detail in the respective sections. All the exercises have been completed. The structure of the codebase has remained the same, with the addition of the main.m file which is used to produce the precision-recall curve. There is also the workplace folder which contains all the testing code. The files ***cvpr_compare*** and ***cvpr_extractRandom***, contain a function for each distance method and feature respectively. The examples for each method consist of testing different parameters on the same query image. For the calculation of PCA, Distances, SIFT and K-Means, MATLAB functions were utilized.

Table of Contents

Global Colour Histogram

Method

This descriptor will retrieve images based on the colour distribution of the whole image. It can match images with overall similar colours, but this doesn't mean that the images will have the same context or include the same objects. The histogram is generated by counting the number of pixels of each kind of colour based on the quantization. This feature is relatively invariant to rotation, small scale changes and partial occlusion (Shapiro, 2000).

Quantization

As we change the Quantization value, the results are also affected in a certain way. Through figures 1 to 3, we can see the different results yielded by each quantization level. In Figure 1, where the quantization level is 2, we see that the first 3 images are almost identical as they contain vast amounts of colour green. A small quantization level means that the histogram will not retain as much information about the colours, so similar colours will be grouped together. Increasing the quantization level to 3, Figure 2, yields the best results providing a balance to detail. But in the case of a bigger quantization level, Figure 3 shows that as colours are quantized in greater scale, they become more intricate and grouping of colours becomes less intuitive to the naked eye. In Figure 4 we can see the distance of the 2 first images in the case of higher quantization. In Figure 5 we can see different results of quantization level 3.

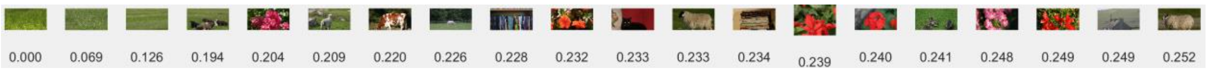


Figure 1. Q = 2.

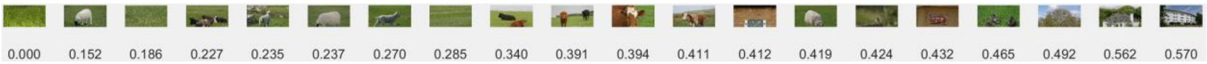


Figure 2. Q = 3.

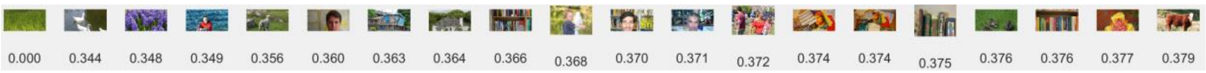


Figure 3. Q = 8.

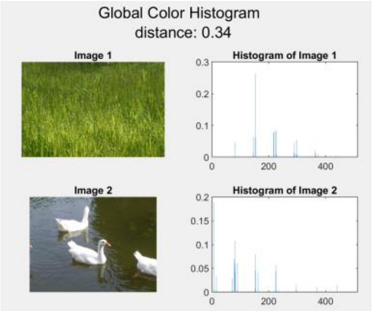


Figure 4. Distance between 2 images and their respective global colour histograms.

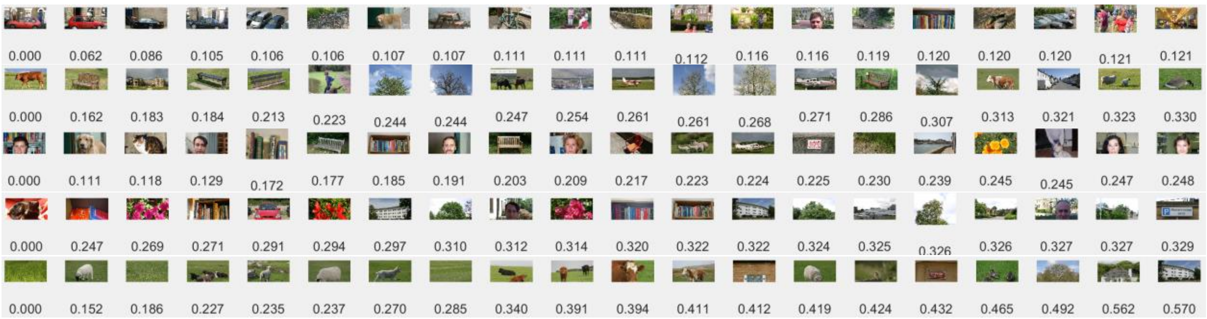


Figure 5. Different query image results with Q = 3.

Experiment Metrics

An initial observation about dataset is that the names of the files have the class encoded, for example images that start with 1 are grass-farm images, 2 are trees and so on. My first thought was to define a class-based similarity, so all my metrics are based on the label of the image and not on the objects in the image as this would require re labelling the dataset. Equation 1 and Equation 2 depict the mathematical formulae used to extract the corresponding metrics. To calculate each part of the formulae, I extracted the class id out of each image in the dataset. The relevant images are the retrieved images with the same class as the query image. The precision can be described as the number of relevant images retrieved divided by the total number of images retrieved while the recall is the number of relevant images retrieved over the number of images in the dataset belonging to the query image class. The precision-recall curve was calculated by retrieving image from top 1 to top 591 with an increment of 25. In the beginning we see a precision score of 1 but very low recall as the retrieved image is the query image and it retrieved 1 out of 28 or 30 images, depending on the class. In the end we get a recall score of 1 as all images were retrieved and a low precision score.

$$precision = \frac{relevant\ images}{amount\ of\ images\ retrieved}$$

Equation 1. Precision formula.

$$recall = \frac{relevant\ images}{amount\ of\ images\ belonging\ to\ the\ query\ image\ class}$$

Equation 2. Recall formula.

Local Colour Histogram

Method

The local colour histogram feature is derived from the global colour histogram features but instead of being invariant to the position of the objects, it focuses on the position of colours in the image by applying a grid on the images and splitting it. After the image has been split into sub images the global colour histogram feature is applied to every one of them. This way we have a colour histogram by region which takes space into account. For this feature a grid function had to be implemented that splits the images into sub images and extracts features from these. The problem that arose with the grid was that the dataset consisted of different sized images, varying both in the Y and X axes. All the images were resized to 210x210 pixels, and a 3x3 sub image grid was applied, meaning each sub image had a size of 70x70 pixels. In Figure 6, different grids were applied such as 3x3, 5x5 and even 10x10 but it did not produce good results.

Quantization

When applying a colour histogram feature on the sub images of the original image we can see that the quantization level that yields the best results in terms of colour distribution is 4 with a grid of 3x3. In Figure 7 two examples are depicted.

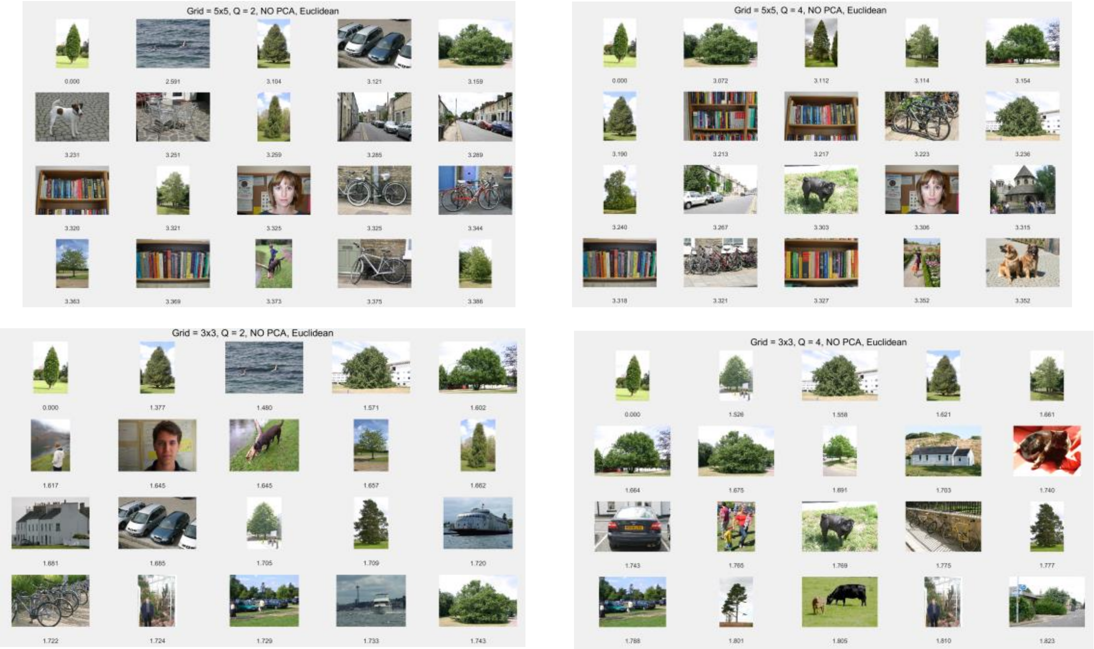

Grid = 5x5, Q = 2, NO PCA, Euclidean
Grid = 5x5, Q = 4, NO PCA, Euclidean
Grid = 3x3, Q = 2, NO PCA, Euclidean
Grid = 3x3, Q = 4, NO PCA, Euclidean

Figure 6. CBIR results based on grid and quantization change.

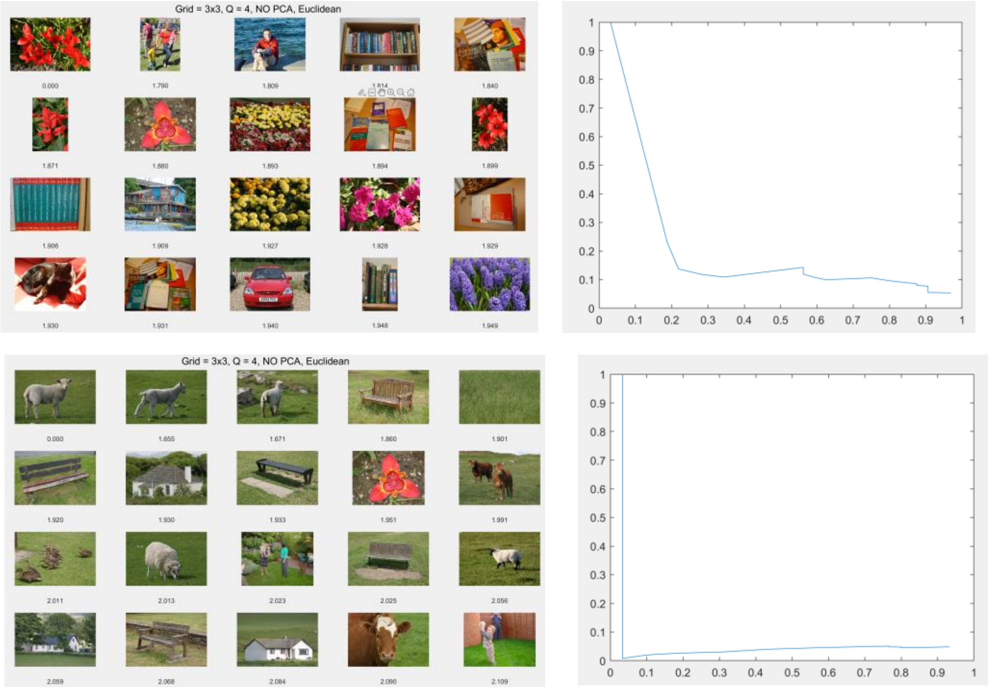


Figure 7. CBIR best results with 3x3 grid and level 4 quantization with PR curves.

Local Edge Histogram

This has a similar implementation as the local colour histogram regarding the image resizing, as a grid must be used as well. The grid that was used is 3x3 with a 70x70 pixel block. The initial number of bins was 20 per block so 180 in total as all the histograms were concatenated to produce the final feature. It is observed that an increase of the number of bins by 24 produces the same or higher metric results; total number of bins **44/block * 9 = 396**. In the case of the **tree** class, Figure 8, while the metrics improved, the irrelevant images also improved in the sense that instead of a photo of bikes with no trees, we get a photo of a street which contains some trees. When the number of bins is increased the final feature can hold more information about the orientations, which is preferred in the case of trees as the leaves have a huge variety of edge orientations just like grass or rough surfaces. On the contrary, in the case of a bench the image, Figure 9 , contains fewer kinds of edge orientations that's why when we decrease the total number of bins to **10/block * 9 = 90**, we have a more efficient image retrieval algorithm even by a little. The airplane class performed the best with 80% precision, Figure 10. We can also see the precision-recall curve.

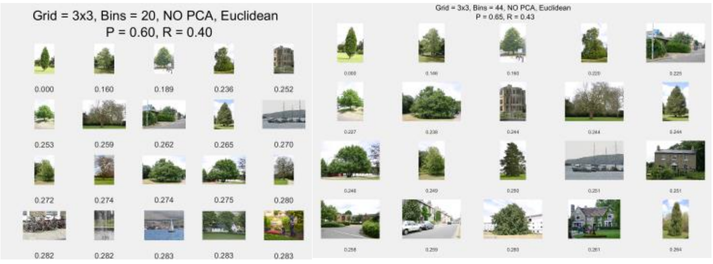


Figure 8. Tree image results with different levels of quantization.

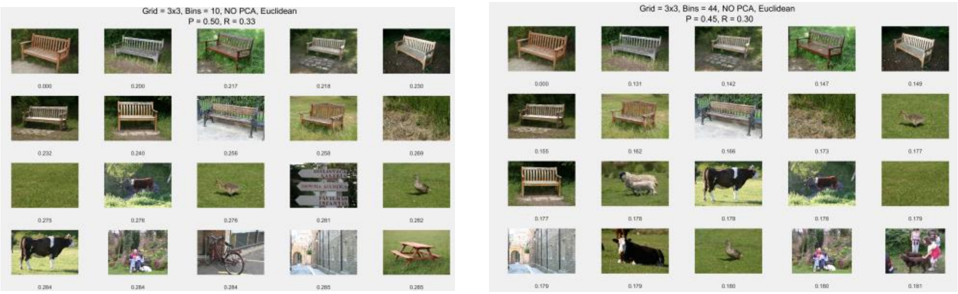

Grid = 3x3, Bins = 10, NO PCA, Euclidean
P = 0.50, R = 0.33
Grid = 3x3, Bins = 44, NO PCA, Euclidean
P = 0.45, R = 0.30

Figure 9. Bench image results with different levels of quantization.

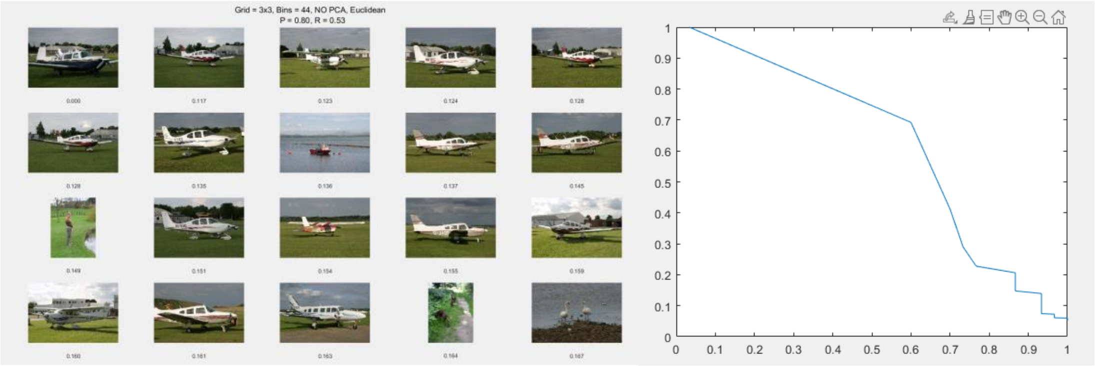


Figure 10. Best Local Edge Histogram results.

PCA

The purpose of PCA (Andrzej Maćkiewicz, 1993) is to reduce the dimensions of the features for compression issues and to battle the curse of dimensionality (Keogh, 2017). New linear combinations are created based on the Eigen vectors and values. Figure 11 shows the projection of all the local edge orientation histogram features down to 2 dimensions, where we can see the distinct group of photos that can be easily separated. The classes of faces and books have a small overlap as they usually have a bookshelf behind each face. If we calculated the precision based on the content of the images the new precision would reach **95%**. From the results depicted in Figure 12 we can conclude that the PCA decreases overall performance on more complex images like trees, but when trying to retrieve simpler images, we have improved performance. We can also the PR curve for the book class.

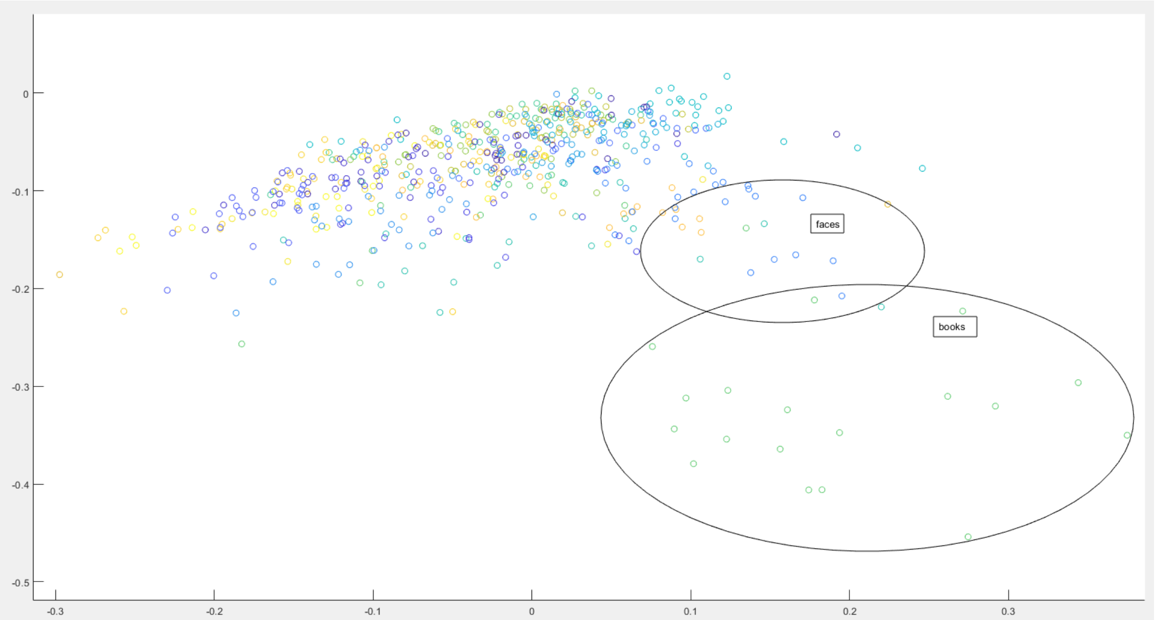


Figure 11. PCA projection of features to 2 dimensions.

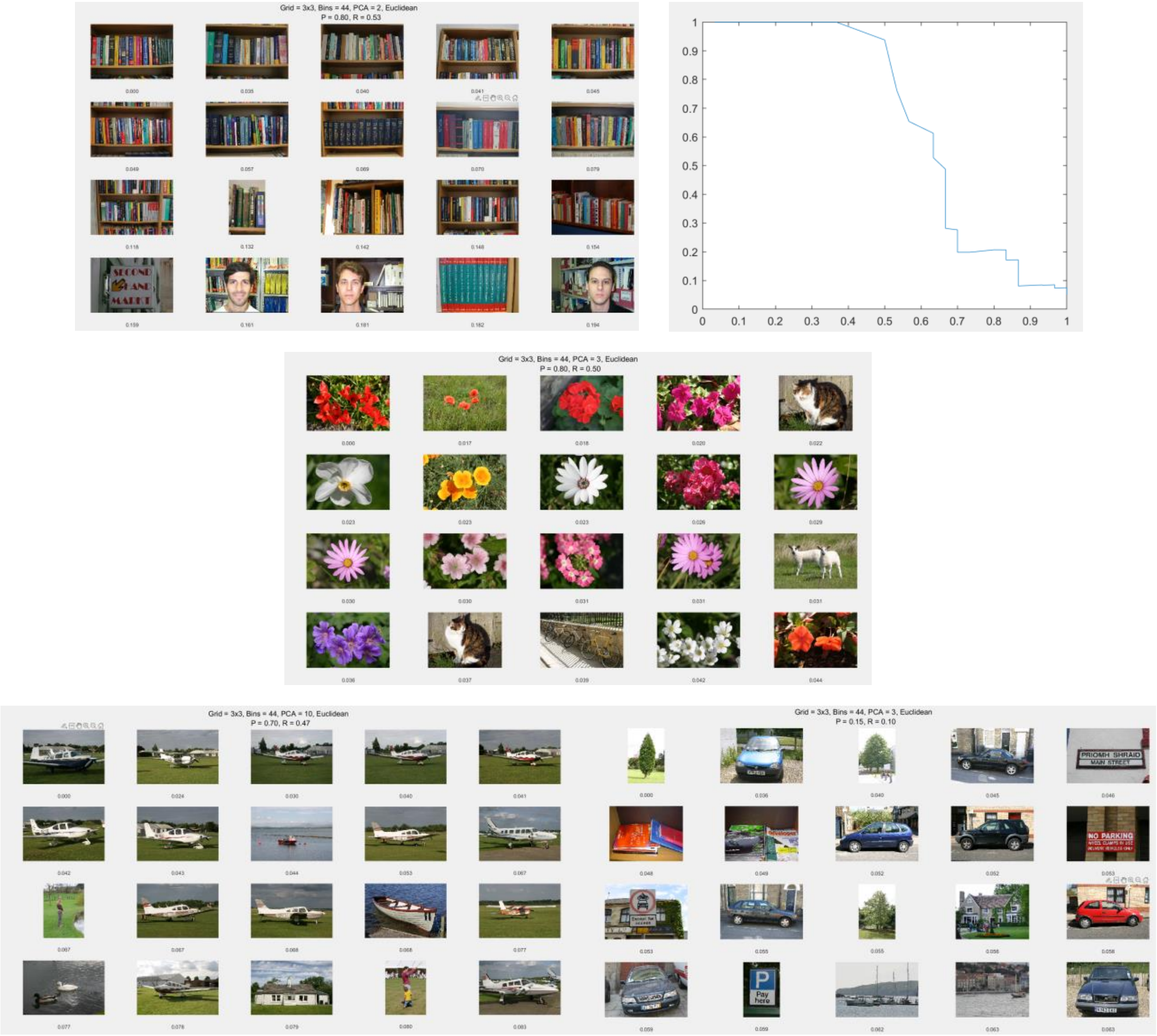


Figure 12. PCA results.

Mahalanobis Distance

The Mahalanobis distance formula between two points as mentioned in (ALLAN DE MEDEIROS MARTINS) is given in the as the Difference of the two points relative to the Covariance matrix as seen below. In general, after some experimenting it seems that the Mahalanobis distance doesn't improve the performance of the system.

$$MD(p_1, p_2, C) = (p_1 - p_2) * C^{-1} * (p_1 - p_2)^t$$

Equation 3. Mahalanobis distance formula between two points.

Distances and Features

The two features I experimented upon the dataset with consisted of the Histogram of Oriented Gradients and the Gray-Level Co-occurrence Matrix, while the distances consisted of L1, L3 and the Chebyshev.

Histogram of Oriented Gradients (HOG)

This feature utilizes the image's gradient orientations to produce a localized histogram for each block of the grid. The idea proposed in (Dalal, 2005), is that the distribution of local intensity gradients can be overall good features when the aim is to describe the contents of an image. For the extraction of this feature the built-in MATLAB function "extractHOGFeatures" was utilised while cropping the features to maintain homogeneous lengths. We must note that this feature is known to perform really well on pedestrian tracking, mostly in an upright position, while the images also had a very specific ration and grid. More specifically the images had a ratio of 2:1. For this reason we are not expecting it to yield the same performance metrics.

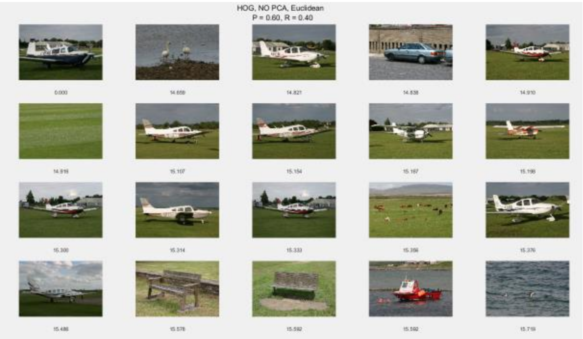


Figure 13. HOG performance with Euclidean distance method.

Gray-Level Co-occurrence Matrix (GLCM)

As described in (Bino Sebastian V1) the GLCM is an early texture feature proposed in 1973. This feature creates a cooccurrence matrix containing the number of occurrences of grey pixel pairs which in turn describes the general texture of the image. After calculating the cooccurrence matrix I extracted the following features: Contrast, Correlation, Energy, Homogeneity. To get a better understanding of the features a constant image has the following respective values of [0, NaN, 1], based on the (MATLAB, n.d.). My first approach was to weight each feature so that the CBIR system performs better but the optimal weights could not be reached.

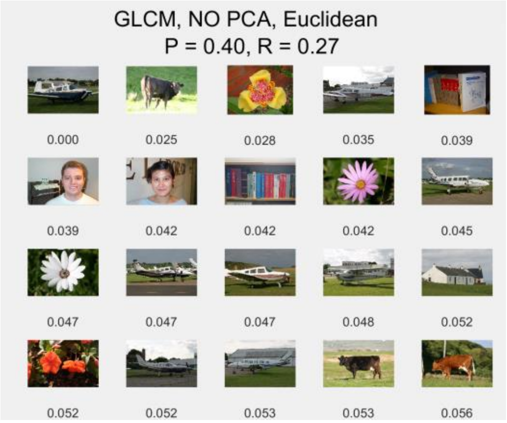


Figure 14. GLCM Features Performance with Euclidean distance method.

Distances

The distances I experimented with, were the Chebyshev, the Manhattan and the L3. Out of them the performance varied based on the class of the image but for the best performing class the L3 distance method boosted its' performance as seen in Table 1.

Distance Method	Precision
Chebyshev	55%
L1	80%
L3	85%

Table 1. Performance of Local Edge Histogram based on Distance Method.

Bag of Visual Words / SIFT

As proposed in (Gabriella Csurka, 2004), the initial feature for this algorithm should be invariant to image transformations, lightweight and yet rich with information. A feature that meets these criteria is the SIFT descriptor which is invariant to scale and rotation. To calculate the SIFT descriptors which are the building block of Bag of Visual Words, the built-in MATLAB function was used which produces the SIFT descriptors of the image. The SIFT points are multiple 128-dimensional points of an image describing key points. Having produced the SIFT features they are stacked and fed into the MATLAB K-Means function which creates 200 clusters based on all the observations of all the images. After the calculation of the clusters a 200-bin histogram is created for each image becoming the final feature vector. To calculate the histogram, each SIFT point of an image is assigned to the closest cluster and thus increasing the respective bin's count by one. At this point the K-Means algorithm is being applied on a 128-dimensional space, which takes time and may suffer from the curse of dimensionality. That is why I applied dimensionality reduction down to two dimensions on the SIFT points and calculated the BOVW. The performance worsened with a precision drop of **20%** as depicted in Figure 15. Note that without the application of PCA the K-Means clustering method could not converge. The number of bins remains the same with the difference that they consist of 2-dimensional points, instead of 128-dimensional points.

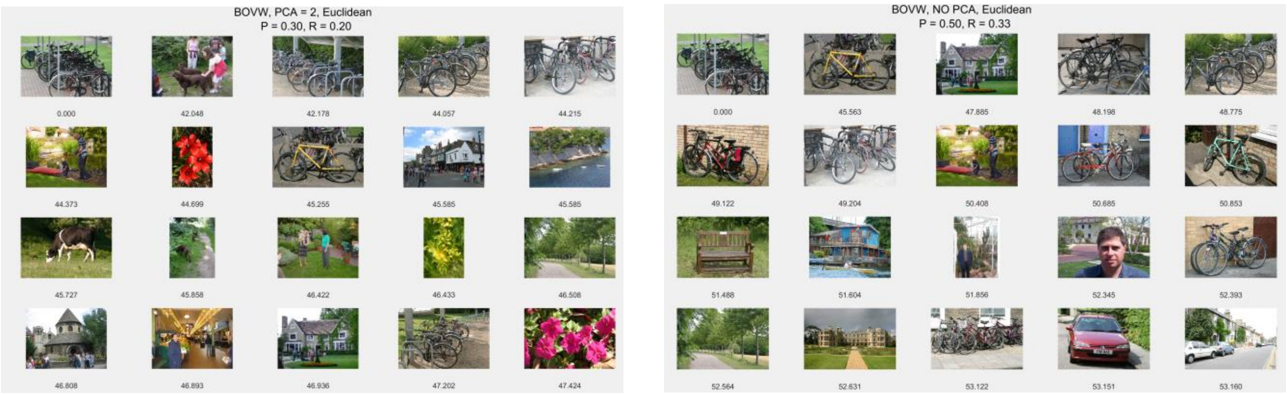


Figure 15. BOVW with PCA and without PCA respectively.

SVM

The SVM includes two approaches, binary and multiclass classification. For the binary classification the classes that were selected were sheep and cars. It is observed that the SVM can precisely differentiate between the two classes as shown in Figure 16. But when experimenting with books and faces we get slightly a worse performance as explained in section PCA. In the case of multiclass we can see that the models do not perform well even after hyperparameter tuning as depicted in Figure 17. For the multiclass SVM I used the **fitcecoc** MATLAB function which includes multiple binary SVM classifiers.

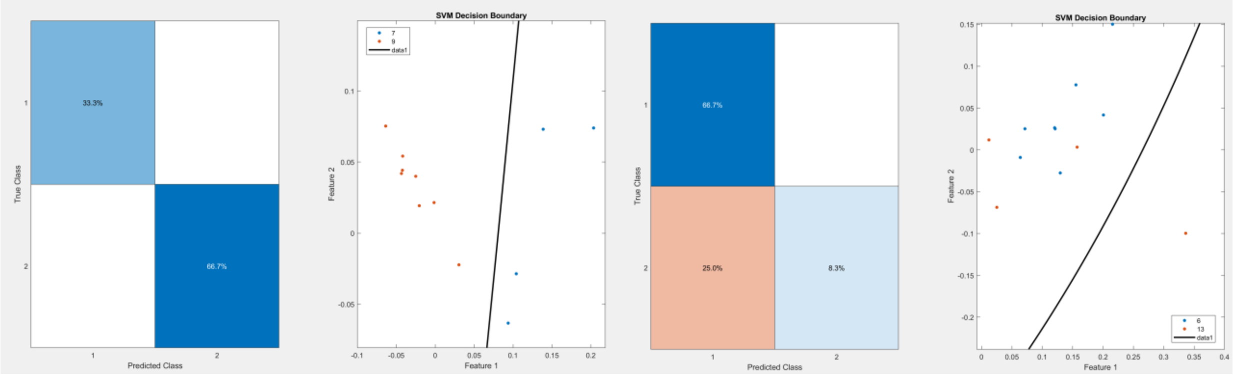
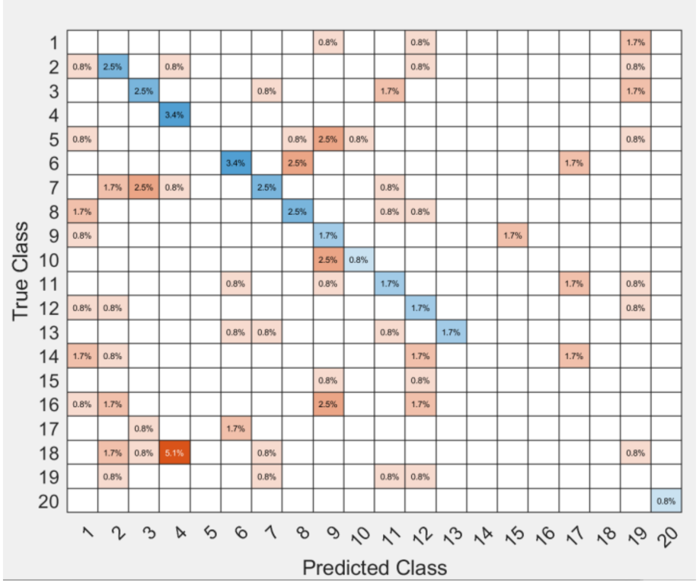


Figure 16. Binary SVM on sheep-cars and books-faces respectively.



True Class \ Predicted Class	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1									0.8%			0.8%							1.7%	
2	0.8%	2.5%		0.8%								0.8%							0.8%	
3			2.5%				0.8%				1.7%								1.7%	
4				3.4%																
5	0.8%							0.8%	2.5%	0.8%									0.8%	
6						3.4%		2.5%									1.7%			
7		1.7%	2.5%	0.8%			2.5%				0.8%									
8	1.7%							2.5%			0.8%	0.8%								
9	0.8%								1.7%						1.7%					
10									2.5%	0.8%										
11						0.8%			0.8%		1.7%						1.7%		0.8%	
12	0.8%	0.8%										1.7%							0.8%	
13						0.8%	0.8%				0.8%		1.7%							
14	1.7%	0.8%										1.7%					1.7%			
15									0.8%			0.8%								
16	0.8%	1.7%							2.5%			1.7%								
17			0.8%			1.7%														
18		1.7%	0.8%	5.1%			0.8%												0.8%	
19		0.8%					0.8%				0.8%	0.8%								
20																				0.8%

Figure 17. Multiclass SVM.

Difficulties

Some of the difficulties I encountered during the coursework, was getting an intuitive understanding of the SIFT descriptors and how to use them in the Bag of Visual Words algorithm. After reading (Lowe, 2004) and the lecture slides I understood the basic concept of SIFT descriptors and was able to implement BOVW.

Conclusions

The best method for the image retrieval system proved to be the Local Edge Histogram with the L3 distance for the airplane class, and Euclidean distance for the rest, and more specifically with 44^3 bins. I expected BOVW to perform the best, because it used the SIFT descriptors, but on the contrary, it had a medium performance. Furthermore, the quantization levels play a significant role regarding colour and edge quantization as results may vary. Finally, the SVM performed well on classes that when visualized in the 2-dimensional space were completely separated, while on classes like books and faces it committed some miss-classifications.

Bibliography

ALLAN DE MEDEIROS MARTINS, A. D. (n.d.). Comparison between Mahalanobis distance and Kullback-Leibler divergence in clustering analysis.

Andrzej Maćkiewicz, W. R. (1993). *Principal components analysis (PCA).*

Bino Sebastian V1, A. U. (n.d.). GREY LEVEL CO-OCCURRENCE MATRICES:. *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.2, No.2, April 2012*.

Dalal, N. a. (2005). Histograms of oriented gradients for human detection., (pp. 886-893 vol. 1).

Gabriella Csurka, C. R. (2004). Visual Categorization with Bags of Keypoints.

Keogh, E. M. (2017). *Curse of Dimensionality, Sammut, C., Webb, G.I. (eds) Encyclopedia of Machine Learning and Data Mining.* Boston: Springer.

Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints.

MATLAB. (n.d.). *graycoprops (Properties of gray-level co-occurrence matrix (GLCM))*. Retrieved from MATLAB: https://www.mathworks.com/help/images/ref/graycoprops.html#d126e126580

Naimeh Sadat Mansoori, M. N. (n.d.). Bag of visual words approach for image retrieval using color information. *Electrical Engineering (ICEE), 2013 21st Iranian Conference on.*

Shapiro, L. (2000). *Computer Vision.* Washington.