Final Project

Contextless Object Recognition with Shape-enriched SIFT and Bags of Features

Author:
Marcel Tella Amo

Supervisors:
Dr. Matthias Zeppelzauer
Dr. Xavier Giró-i-Nieto

28th August 2014

1 Abstract

Currently, there are highly competitive results in the field of object recognition based on the aggregation of point-based features [4, 26, 5, 6]. The aggregation process, typically with an average or max-pooling of the features generates a single vector that represents the image or region that contains the object [7].

The aggregated point-based features typically describe the texture around the points with descriptors such as SIFT. These descriptors present limitations for wired and textureless objects. A possible solution is the addition of shape-based information. [9, 6, 2, 12]. Shape descriptors have been previously used to encode shape information and thus, recognise those types of objects. But generally an alignment step is required in order to match every point from one shape to other ones. The computational cost of the similarity assessment is high.

We purpose to enrich location and texture-based features with shape-based ones. Two main architectures are explored: On the one side, to enrich the SIFT descriptors with shape information before they are aggregated. On the other side, to create the standard Bag of Words [7] histogram and concatenate a shape histogram, classifying them as a single vector.

We evaluate the proposed techniques and the novel features on the Caltech-101 dataset.

Results show that shape features increase the final performance. Our extension of the Bag of Words with a shape-based histogram(BoW+S) results in better performance. However, for a high number of shape features, BoW+S and enriched SIFT architectures tend to converge.

keywords

SIFT, interest points, object candidates, segmentation, Bag of Words, shape coding, object detection, textureless objects, wired objects.

Contents

List of Figures

2 Acknowledgements

First of all, I would like to thank Matthias and Xavi for all the good advice that they have given to me. Not only all advices but the organization has been perfect as well. Despite the geographical difficulties that we have had some times, we have been still able to maintain the project always alive and running. For me, it has been an amazing experience.

Carles Ventura has also been a very active part of the project, despite not being my advisor, he has attended to some of the meetings and has strongly contributed to some ideas. His help has been very valuable.

This project would not have been possible also without the help of Albert Gil and Josep Pujal, who have helped me a lot with computations and technical difficulties.

My girlfriend Raquel came with me to Vienna and she gave me all the best support that I could ever have.

I would like to say that we had a very good welcome to Vienna by Matthias and all people in the department.

And finally, what to say about our family there, Renata, Igor and Filip. They have shown us everything, and we could not be more thankful for all the little things that they have taught us and the good times there. I could never find the correct words to express how amazing these days were.

3 Motivation

The main goal of the project is the study of the addition of shape information to the popular Bag of Words[7] aggregation scheme. This pipeline consists in creating a visual vocabulary by quantizing local descriptors. Once the vocabulary is created, a histogram is computed by assigning new points to the closest codeword in the vocabulary. This histograms then are used as input for a classifier.

At the moment of the writing, there is not much work on shape coding by combining segmentation techniques and interest points [2, 6, 9, 16, 17]. However, there is a lot of work done on segmentation and object candidates [13] as well as in interest points [4, 26, 5, 24, 26, 27, 25]. The information shared by both can be used to codify the shape, by taking advantage of the two approaches.

Shape coding is already an explored field [3, 2], but it is generally used in conjunction with some kind of alignment. The alignment allows the matching of a shape with other shapes in order to obtain a measure of similarity. It increases the computational cost of the similarity assessment when comparing two shapes.

On the other hand, some authors [6, 11] have explored the enrichment of SIFT descriptors in aggregation architectures that do not require any alignment. Enrichments include, for example, color [6] and spatial coordinates [11], as well as rough shape descriptions through the relative coordinates with respect to a region's bounding box [6]. Their results have shown an increase in performance that motivated our work enriching SIFT features with accurate shape information of regions that represent objects.

We investigate two principally different approaches (architectures): The enrichment of descriptors with additional features before the creation of the vocabulary and the enrichment of Bag of Words histograms with other shape-based histogram representations.

Firstly, the enrichment of the SIFT descriptor with shape information is simply to add extra features at the end of the 128-dimensional SIFT descriptor. For the study, a wide variety of shape features have been considered to see if the performance can be increased by using this aggregation scheme.

When the SIFT descriptor is extended, the size of the Bag of Words is a design parameter. Experiments have been performed in order to see if by increasing the size of the feature vector, there is the need to increase the size of the visual vocabulary as well.

Secondly, two different aggregation schemes have been considered. They are the Bag of Words [7] approach as well as the Second Order Average Pooling [6]. The main difference is that Second Order Average Pooling uses shape matching and features do not need to be quantized.

For the second architecture we append a shape descriptor to the BoW histogram. An important aspect in this context is the relation of the dimensionality of both histograms of features. The question is what happens if we increase the size of one of them, reducing then, the effect of the other histogram.

This thesis report is structured in the following sections. Section 3 presents the requirements of the project. The state of the art is presented in Section 4. Then, an overview of the working flow of the projects is shown in Section 5.
In Section 6, the design of all new features is presented, followed by the development in Section 7 which shows practical issues that have been explored. To conclude the work, evaluations, results and conclusions are presented in Sections 8 and 9. Additionally, a Section for the future work is also presented in the last section.

4 Requirements

The basic requirements of this project are:

1. Design a shape feature that can be used in an aggregated framework, with no need of matching or alignment.
2. Study how to enhance object recognition using aggregated features.
3. Study the limitations of shape coding when using a state of the art segmentation.
4. Analyse the implication of the vocabulary size when the length of the feature vector grows.
5. Study the relative importance of shape features when combined with BoW histograms.
6. The proposed features should be at least scale, rotation and translation invariant. If it is possible, flip invariant as well.

An important point to consider is that this research will try to maximize the accuracy of the results, even at the expense of higher computational solutions.

Results will be obtained from a publicly available scientific dataset that will allow the reproduction of the experiments. These experiments should be comparable to as many state of the art publications as possible. The resulting source code of the used software will be made available to the scientific community to allow for its validation.

5 State of the art

Many approaches of object recognition tasks have been explored and this chapter is an overview of the most important ones in the scope of this project.

5.1 Shape

To take information from an object's shape, two tasks are required: Firstly, the ability of extracting regions that accurately represent an object. These type of algorithms are called object candidate algorithms. Secondly, there is the need of codifying the shape in well distinctive features.

5.1.1 Object Candidates: CPMC and MCG

An object candidates algorithm is the first step in the state of the art for object recognition [18, 19, 6] which has to detect and locate the object in the image. In [13], Constrained Parametric Min-Cuts are applied in order to identify object candidates and gives a higher score to the ones that are more likely to be real candidates.

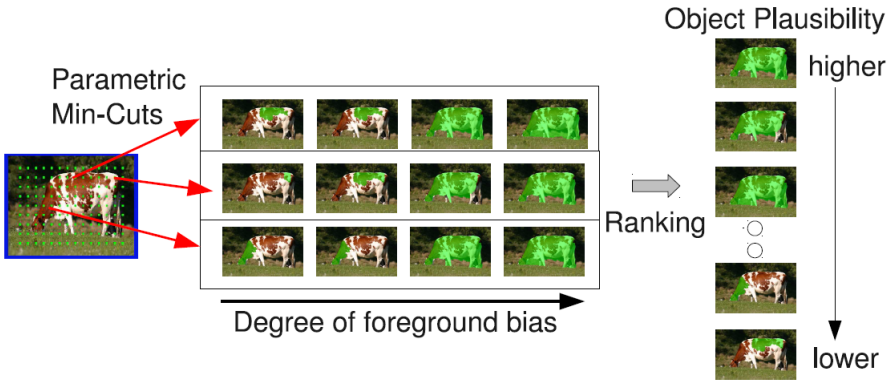


Figure 1: Constrained Parametric Min-Cut algorithm. Figure taken from [13].

Given a set of pixels in an image, a selection of them are hypothesised to belong to the foreground or background. At the beginning, a 5x5 grid of seeds is spread all over the image. Later, a prediction of the real value is made by turning the problem into a Graph Cut problem [14] which is solved by minimising a cost function defined with the goal that real foreground pixels have the minimum cost.

Multi-scale Combinatorial Grouping [20] is an approach for bottom-up hierarchical segmentation for object candidates generation. The main idea is to use multi-scale information to group regions in different scales into high-accurate

object candidates in a very efficient way in the scale space. The output of the algorithm will be then used to perform very accurate segmentations as input for our shape-based features.

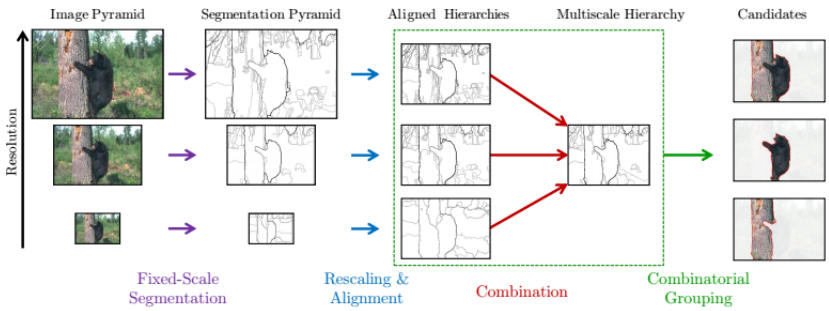


Figure 2: Figure taken from [20]. MCG candidate selection

5.1.2 Shape Descriptors

This section reviews some techniques that transform the region into a set of features that clearly capture the shape of the object. Given that human beings can recognize an object when only the shape is given, these descriptors become very important.

5.1.2.1 MPEG-7 Visual Shape descriptors

In shape coding, the MPEG-7 Visual Shape descriptors[3] are an effort to standardize its data format to facilitate their exchange and inter-operability. The section of 2D object or region descriptors is based in two different approaches: The Region-Shape descriptor and the Contour-Shape descriptor. The *Region-Shape descriptor* captures the distribution of all pixels inside the region.

The *Contour-Shape Descriptor* is based on the Curvature Scale Space technique (CSS) to represent the contour. This technique is based on the equidistant subsampling of the contour starting form an arbitrary point, obtaining then a set of x and y coordinates.

A Gaussian function is convolved with the parametric representation of the shape. CSS decomposes the resulting function into concave and convection parts by getting the zero-crossing points. The set of parameters result in a descriptor which has invariance to rotation, uniform scaling and translation. Figure 3 shows an example of CSS representation.

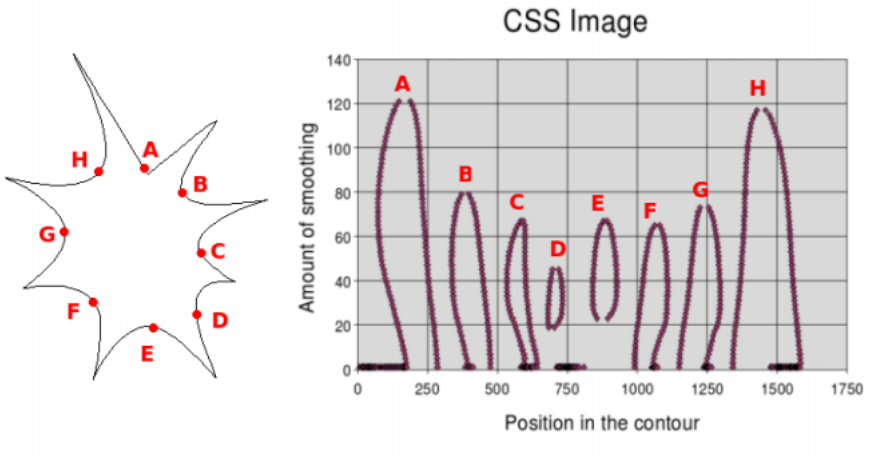


Figure 3: CSS technique. Figure taken from [22].

Eccentricity and circularity are also computed and added to the descriptor:

$$circularity = \frac{perimeter^2}{area}$$

$$eccentricity = \sqrt{\frac{i_{20}+i_{02}+\sqrt{i_{20}^2+i_{02}^2 i-2i_{20}i_{02}+4i_{11}^2}}{i_{20}+i_{02}-\sqrt{i_{20}^2+i_{02}^2 i-2i_{20}i_{02}+4i_{11}^2}}}$$

$$i_{02} = \sum_{k=1}^{N}(y_k - y_c)^2, i_{20} = \sum_{k=1}^{N}(x_k - x_c)^2, i_{11} = \sum_{k=1}^{N}(x_k - x_c)(y_k - y_c)))$$

Where N is the number of samples in the contour (x_k, y_k) and (x_c, y_c) are the coordinates of the center of mass of the region.

Then, the curvature function is calculated as well:

$$K(u,\sigma) = \frac{X_u(u,\sigma)Y_{uu}(u,\sigma)-X_{uu}(u,\sigma)Y_u(u,\sigma)}{(X_u(u,\sigma)^2+Y_u(u,\sigma)^2)^{\frac{3}{2}}}$$

where

$$X(u,\sigma) = x(u) * g(u,\sigma)), Y(u,\sigma) = Y(u) * g(u,\sigma))$$

$$X_u(u,\sigma) = x(u) * g_u(u,\sigma)), X_{uu}(u,\sigma) = x(u) * g_{uu}(u,\sigma))$$
$$Y_u(u,\sigma) = y(u) * g_u(u,\sigma)), Y_{uu}(u,\sigma) = y(u) * g_{uu}(u,\sigma))$$

and $g_u(u,\sigma)$ is a 1-D Gaussian kernel with deviation σ. Then, for different σ values, zero-crossing values are obtained and kept in the descriptor with the corresponding σ value.

Auxiliary shape descriptors such as area descriptor, bounding box descriptor, and additional descriptors can be found in [22, 21, 23].

5.1.2.2 Shape Context

The shape context algorithm [2] takes advantage of the points from the contour of a region and measures the log distance from each point to all other points with the purpose of building a 2D-histogram per point. In the example in Figure 4, 5 log distances and 12 angles have been used to create the histograms.

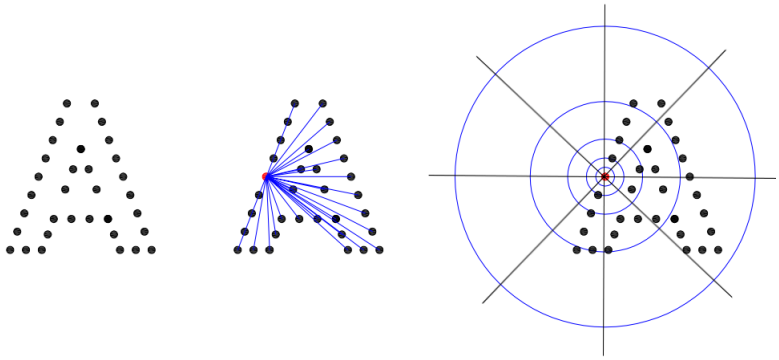


Figure 4: Shape Context algorithm.

As shown in Figure 4 in the center image, a point in the quantized shape is chosen to calculate the distance with all other points. This is done for all points, creating then the following histograms per point.

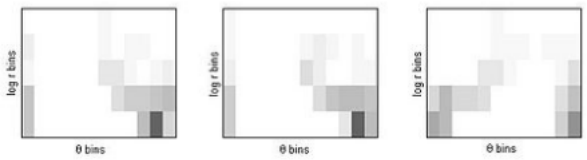


Figure 5: 2D histograms, log distance vs angle.

Then, a matching algorithm is used to be able to assess the similarity between histograms. It is based in bipartite matching graphs [1] where every shape can be compared to other shapes reporting then how similar they are. With the 2D histograms, the χ^2 test statistic is computed as:

$$C_{ij} = C(p_i, p_j)) = \frac{1}{2} \sum_{k=1}^{K} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}$$

where $h_i(k), h_j(k)$ denote the K-bin normalized histogram at p_i, p_j.

The similarity is estimated by minimizing a cost function between all pairs of points (pairs of histograms).

$$H(\pi) = \sum_i C(p_i, q_{\pi(i)})$$

A very important point in this algorithm is that the distances from point to point are directly tracing straight lines from point to point even when the lines cross outside the borders of the region.

5.1.2.3 Inner Shape distance

The main difference between the Shape Context approach [2] is that here, the Euclidean distance is replaced by the inner distance [12]. It is defined as the shortest path between landmark points within the shape silhouette. In this way, articulation invariance is achieved. An easy example could be a hand: The distance from a point in a finger to a point in the consecutive finger. If both fingers are joint or separate, this distance goes through the union, called junction, and the inner distance results similar.

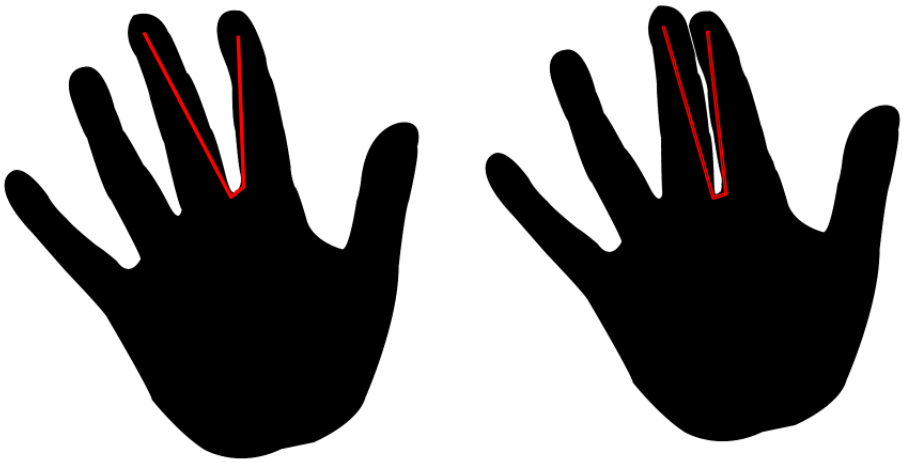


Figure 6: The inner distance (in red) between two points is very similar when the movement involves articulations.

5.2 Interest points

An interest point can be defined as a point which has something distinctive with respect to the surrounding ones. Interest point-based image representations consist in (*i*) the interest point detectors and (*ii*) the interest point descriptors.

An interest point detector tries to find a set of the most salient points in an image. One of the first detectors is the Hessian detector [25] which uses a multiple scale iterative algorithm to spatially locate and select scale and affine invariant points. Then, the Difference of Gaussians was proposed in [24] to approximate the Hessian matrix for faster computations.

A very efficient implementation is the Speed Up Robust Features (SURFT) interest point detector. Other examples of keypoint detectors are the Harris

Figure 7: Sparse SIFT, the location of the points have been computed to select the most important points in the image.

detector and the SUSAN detector.

Interest point descriptors are used to describe the neighbourhood interest points. One of the most used descriptors is the Scale Invariant Feature Transform [24]. It works as follows: Firstly, it uses the image gradients and orientations around the point location using its scale and weighted with a Gaussian function to give less relevance to points far from the center. Later, 8-bin orientation histograms are created over a sample grid of 4x4. Therefore, each keypoint is represented by 128 features (4 rows x 4 columns x 8 bins in the histogram). Finally, the feature vector is normalized to reduce illumination changes.

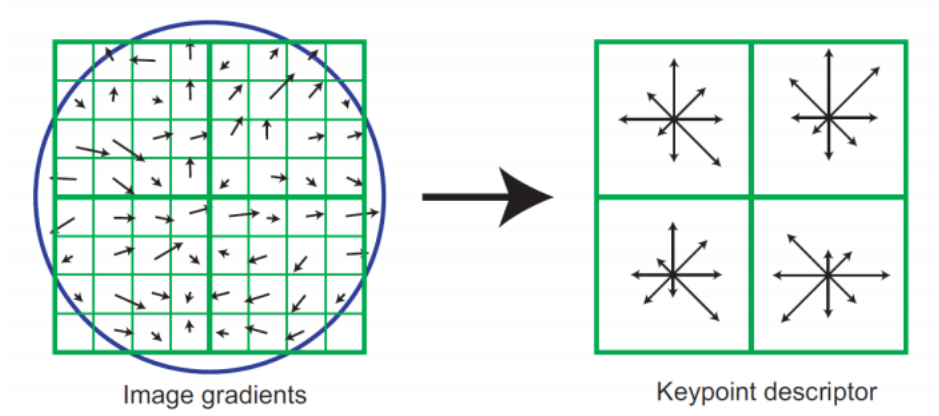


Figure 8: figure taken from [24]. Gaussian window in blue, this image contains a 2x2 grid of histograms in the right from 4 sets of 4x4 grids in the left.

The SIFT descriptor reflects the textured information and we are using it in order to get the textured part of the image and combine it with shape descriptors.

Other evolved versions from SIFT can be the Histogram of Gradients(HOG) [27] whose basic idea is that intensity gradients can characterize the object appearance and shape. There SURF descriptor [26] describes a distribution of

Haar-wavelet responses within neighbourhood pixels.

Dense SIFT, however, is a technique where the location of all points is known beforehand by using a regular grid. The SIFT descriptor is applied to those locations. DAISY descriptors have the advantage they are rotation invariant whereas dense SIFT generally is not for the sake of a fast computation.

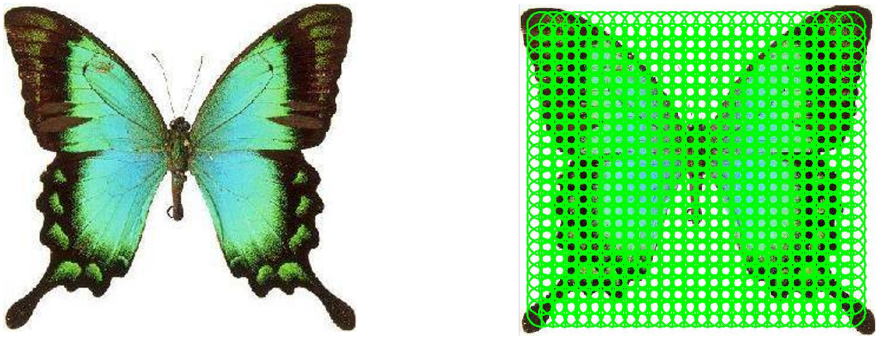


Figure 9: A grid covers all the image and SIFT descriptors are extracted. SIFT detector is not used.

5.2.1 Enrichment of the SIFT descriptor

The idea of extending the SIFT descriptor has been already tried in [6] by using the color, position and aspect ratio of the bounding box containing the object. The features proposed are:

- 4 dimensional aspect ratio features from the bounding box of the region.

$$\left(\frac{f_{ix}+f_{jx}}{w_j}, \frac{f_{ix}+f_{jx}}{h_j}, \frac{f_{iy}+f_{jy}}{w_j}, \frac{f_{iy}+f_{jy}}{h_j}\right)$$

  Where h and w are the dimensions of the bounding box, and i and j are referring to the bounding box space(b) or image coordinates space(f).

- 2 scale features. where s_i, s_j are the scales of the interest points.

$$\beta[\frac{s_i}{w_j}, \frac{s_i}{h_j}]$$

- The average color in three different color spaces: RGB, HSV and LAB.

The classification method used in this work is the second order average pooling, explained in the next subsection.

In [11] an enrichment based on the absolute position of the points has been tried reporting better results. The main goal of this technique is to try to emulate a

spatial pyramid without actually performing it, thus, saving a lot of computation time. The spatial references added at the end of the SIFT descriptors are set in Cartesian coordinates or as in Polar coordinates with respect to the center of the image. Authors claim that the second one works better.

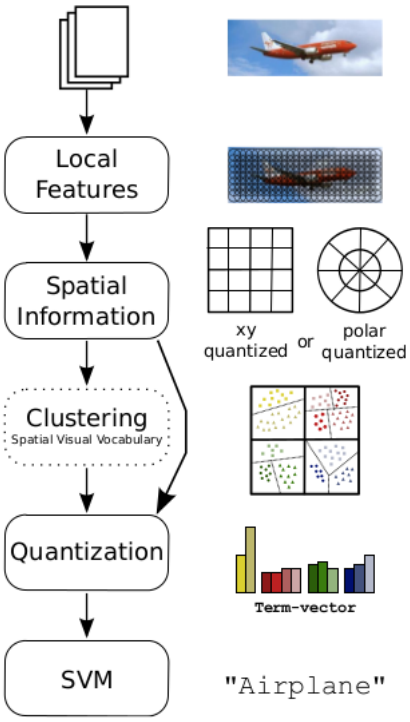


Figure 10: Figure taken from [11]. Scheme of spatial enriching of the SIFT descriptors with the goal of emulating a spatial pyramid.

After adding the spatial information to the local SIFT features, the result is a set of vectors representing descriptors of 128(from SIFT)+ N(spatial) dimensions. In the next step, a quantization of the descriptor is used by performing a clustering with the *k-means* algorithm and resulting into a vocabulary of visual words. The goal of the vocabulary is to allow the representation of images by means of histograms containing the quantized set of words. The last step to perform would be the classification using a SVM.

5.3 Aggregation of points

After showing some features and the way to codify them, the remaining question is how to combine the individual features from a set of points to describe a larger entity such as a region or an image. This will be reviewed in the next sections.

5.3.1 Bag of Words

When it comes to aggregate new features, one of the most extended approaches in object recognition is the Bag of Words [7]. This approach is named by the fact that the point-based descriptors are considered as an unsorted collection of items, in an analogy to how items are randomly placed in a bag. It consists on quantizing local descriptors in K different visual vocabulary words, where K is a design parameter. In order to apply the bag of words on local descriptors it is necessary to quantize the feature space. This partition of the space is typically performed with the K-means clustering algorithm. The centroid of the resulting clusters is named a visual word and the collection of visual words is a visual vocabulary.

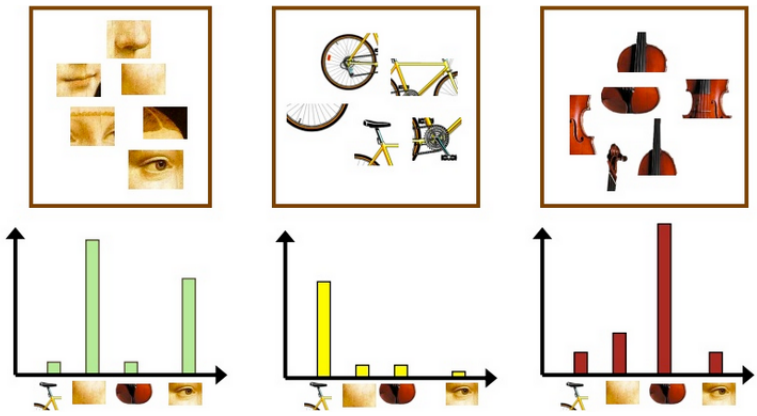


Figure 11: Three examples of visual bag of word histograms.

The main idea of this pipeline is to discover how many of these visual words appear in each region or image by creating histograms, so-called Bags of Words or Bags of Features.

The resulting histograms are typically normalized by the total amount of visual words in the region or image. As a result, this type of aggregation is considered an averaging aggregation.

5.3.2 Second order pooling

In contrast to the Bag of Words pipeline, second order pooling[6] does not need to quantize the descriptors for its aggregation. From the descriptor, a second order matrix is created through the outer product between two enriched SIFT feature vectors. This generates a matrix to represent each point which captures the correlation between the feature components, similarly to a single Gaussian model.

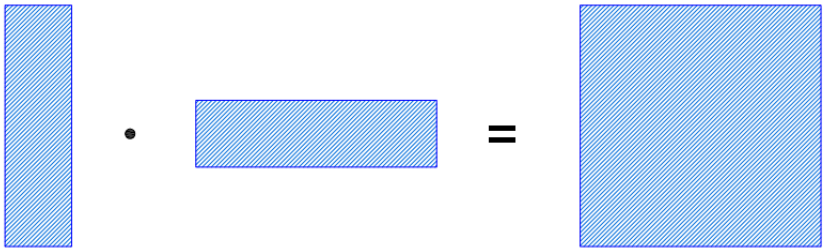


Figure 12: Outer product

This matrices of second generated for each point are aggregated with an average pooling to obtain a single matrix for the whole region or image.

$$G_{avg}(R_j) = \frac{1}{|F_{Rj}|} \sum_{i:f_i in R_j} x_i x_i^T$$

Additionally, the logarithm of each component in the matrix, so that a linear SVM classifier can be used in a later stage of the process.
Finally the algorithm then just considers the upper right diagonal of the matrix as it is symmetric, and scans it to generate a feature vector of dimension 10,000.

6 Working Plan

The organization and working plan is an important part of the success of the project. As we did not know how the flow of the project would be completely from the beginning, the project working plan was being created while the project has been progressing, always looking some weeks ahead.

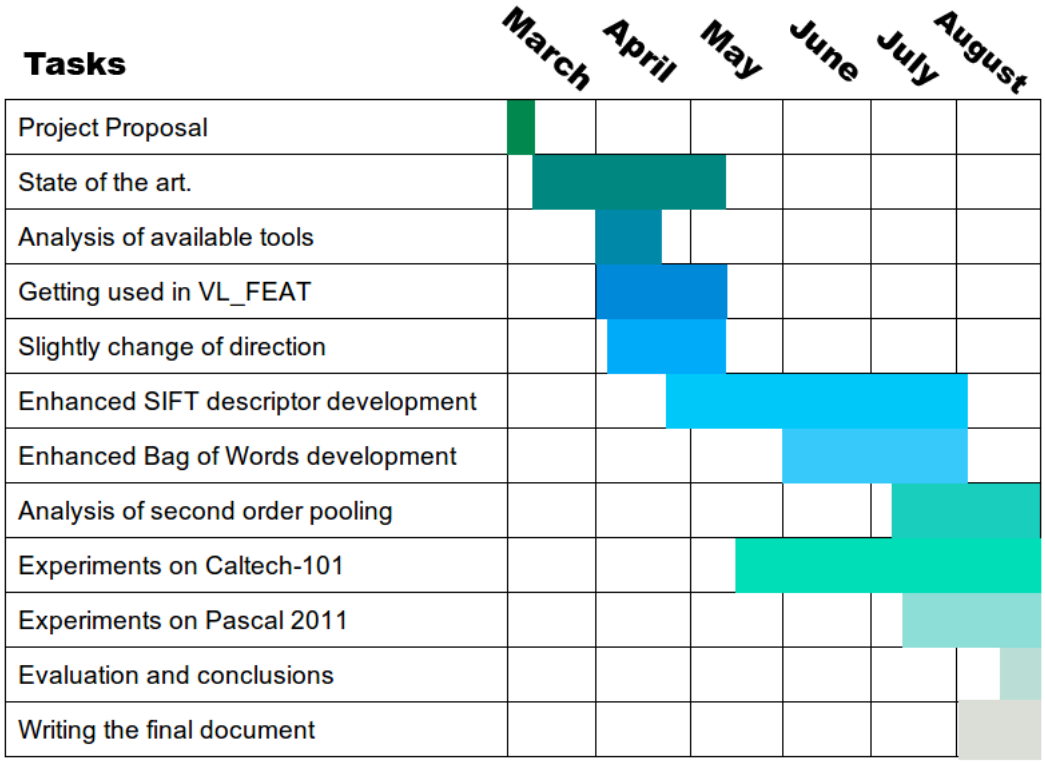
The project has been followed closely with weekly meetings as well as email discussions about the day to day topics that have been appearing.

6.1 Tasks

The tasks to accomplish in the project have been:

1. **Project proposal:** At the starting point, we had some topic to work on, we decided textureless object recognition.
2. **State of the art:** A lot of literature has been revised in order to define a direction for the project.
3. **Textureless object recognition:** There was a slight change of direction, we started to work in enrichment of SIFT features and bags of words of SIFT features. This techniques, as they enrich with shape information, should also work good with textureless objects.
4. **Analysis of available tools:** An overview of all possible libraries and frameworks was made to see which one was fulfilling better our requirements.
5. **Familiarization with the VL_FEAT framework:** Due to the variety of functions that it has, it fits perfect in this study.
6. **Development of the enhanced SIFT descriptor methods:** The first stage of the project.
7. **Development of the enhanced Bag of Words methods:** The second stage of the project.
8. **Experiments and results on Caltech-101:** We use Caltech-101 to do our experiments. With a specific set of training and testing images, later commented, as well as other design parameters that will give us a huge variety of experiments.
9. **Evaluation and conclusions:** The results of the later experiments are analysed here in order to see the strength and weakness of our algorithms.
10. **Writing of the thesis report:** Write the final thesis report in order to be reviewed by Technical University of Catalonia professors for the final presentation.
11. **Presentation of the work:** Presentation of the work to the Technical University of Catalonia.

6.2 Gannt Diagram



7 Design

This section presents the design of the collection of techniques proposed in this work, by combining segmentation and interest point techniques.
All techniques follow the same basic architecture. At first, obtaining a collection of regions based on the MCG object candidates algorithm and, secondly, obtain a set of uniformly sampled SIFT points inside these regions. Each of these points is considered in order to extract the features.

To introduce the following section, an interest point descriptor is needed in order to gather the textured information. However, in some types of objects, interest point detectors and descriptors can have the problem addressed in the next section.

7.1 Uniform sampling vs Sparse Point detectors

In textureless regions, sparse SIFT does not place points. To get information of these regions it is necessary to use the dense version despite a these points will have not much information. In the present work, a dense SIFT will be computed for different scales. The main drawback of this strategy, however, is that a lot of descriptors are being computed, while in sparse SIFT, only the salient points in the image are fetched.

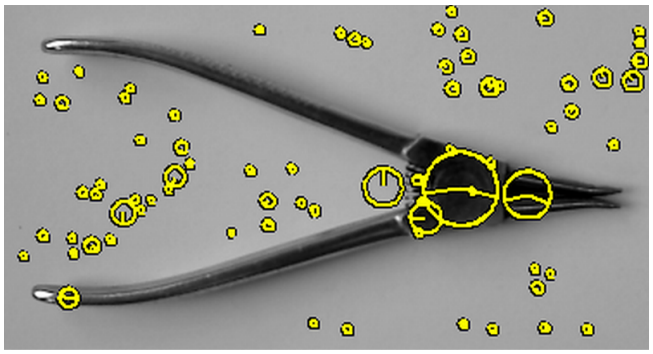


Figure 13: Textureless regions are not covered by the sparse SIFT.

7.2 Shape Descriptors for aggregation

Dense points in conjunction with object candidate segmentations allow the analysis of the points which are inside the region whereas the points outside the region are neglected. We aim to analyse the shape of the region by using this mixed approach. For the sake of brevity and quick references, each technique is denoted with the corresponding acronyms in its title.

7.2.1 Distance to the nearest border (DNB)

This technique measures the distance between the point and the closest point at the contour. The result is a vector of two dimensions for each point in the grid.

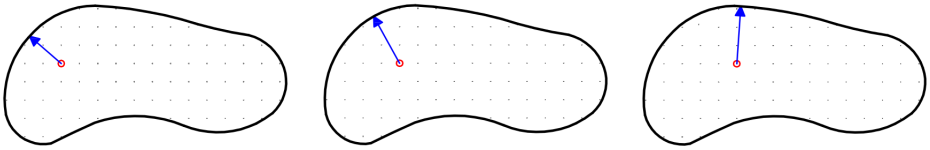


Figure 14: Vectors from a point in the regular grid going to the nearest point at the contour.

By drawing the vectors corresponding to all SIFT points inside the region and computing the distance to the nearest border, an interesting plot appears (Figure 15).

In Figure 15, there are two examples of the graph resulting of plotting a vector from each SIFT point to the closest point in the contour. Red points correspond to the dense SIFT points falling inside the object whereas green points are the closest points in the contour of the image.

In this figure, we can clearly visually distinguish the skeleton of the region as the line inferred by the less dense area, which is telling that somehow, the shape is being codified.

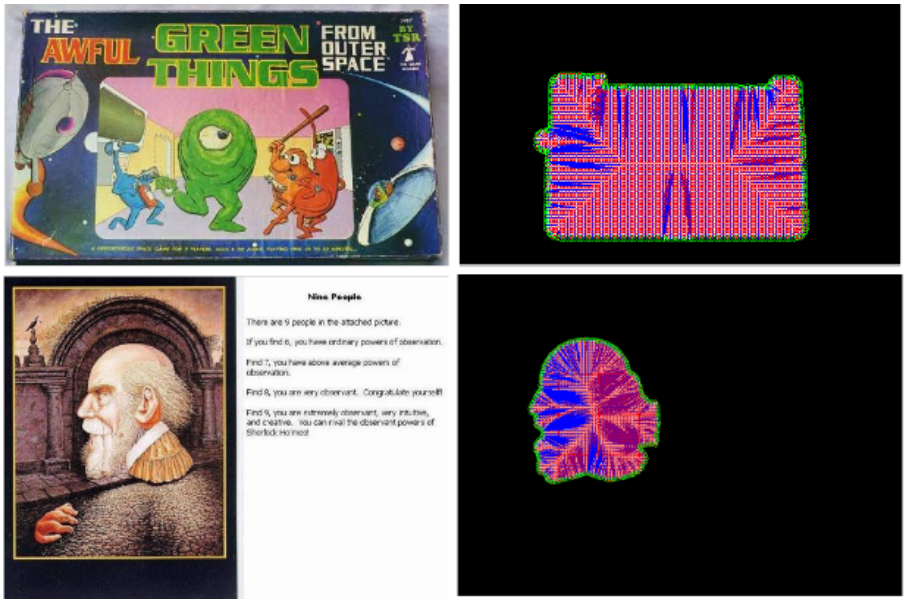


Figure 15: In the left, the original images are shown. In the right, the graphs resulting to to the distance to the nearest borders.

The distance to the closest point in the border defines for each point, the radius of an inscribed circle. Thus, it is a way to describe the size of the immediate surroundings of each point.

A normalization stage is required in order to guarantee that the descriptors is invariant to spatial scale. This means that the same shape with a different area in terms of pixel count will be represented by the same descriptor. This normalization is performed by diving all descriptors by the longest distance found in the region, so that the range of the descriptor becomes [0,1].

$$x' = \frac{x}{max(O)}$$

Where O is the subset containing all vectors for the current region.

7.2.2 Logarithmic distance to the nearest border (LDNB)

In [2], a logarithmic distance has been proposed for shape description. The effect of taking the logarithmic distance is that elongated objects become shorter after the transformation.

$$l = log(d)$$

7.2.3 Distance and Angle to the Nearest Border(DANB)

The above defined DNB descriptor can be extended by considering the angle defined between the line to the nearest point in the border and a reference axis(e.g. the 3 o'clock direction).

If the angle is codified in degrees or radiants, however, there is a point where the difference between similar vectors is big. The case where the vector is pointing at 359º is visually very similar to the one that points to 0º. So, despite graphically, the difference is minimal, and could be said that these vectors are quite similar, the difference in range is huge.

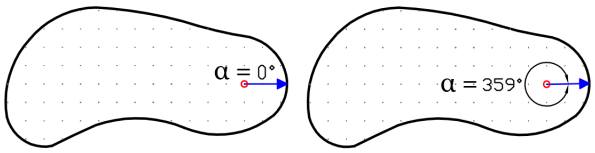


Figure 16: Despite the vector is almost the same, the resulting angle is completely different.

A more convenient solution is to store the angle in two features.

$$[\sin(\alpha), \cos(\alpha)]$$

SIFT points falling exactly on the border, have the modulus equal to zero and the vector distance becomes a single point, and thus, the angle cannot be computed. For this reason, all SIFT points on the contour are discarded.

7.2.3.1 Rotation Invariant Angle To The Nearest Border

To codify the angle an absolute direction is not the best solution. Let's say that the angle is codified with respect to the 3 o'clock direction. Once the image is rotated, the angle for the same object is going to change, and the resulting feature will be different.

A relative coordinate system is created then by taking advantage of the major axis of the region. From that direction, by having a direction and a vector, there are two possibilities as there are two possible(diametrically oriented) unit vectors which may be taken as major axis. The minimum angle between the vector and the direction is chosen so that, if the region is rotated, the angle is going to be exactly the same.

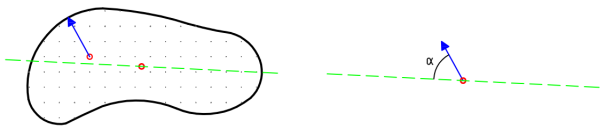


Figure 17: The computation of the minimum angle between major axis and vector direction results in rotation invariance.

In Shape Context [2], the direction is chosen arbitrarily and thus, it is not rotation invariant. The proposed representation in contrast achieves rotation invariance by using the proposed local reference axis.

7.2.4 Distance to the center(DC)

The main idea behind the following approach is to try to relate the location of the points with an intrinsic feature of the region.

It consists on calculating the distance between the centroid of the region and all SIFT keypoints in the grid. Distance and angle could be codified in the same way as in the **distance to the nearest border** approach.

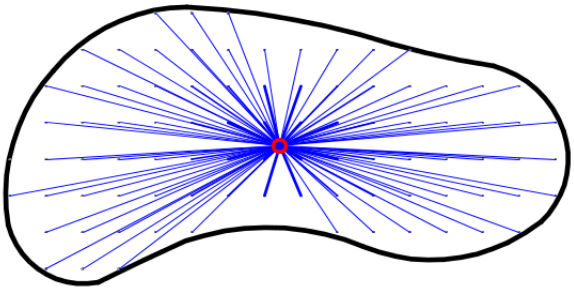


Figure 18: Computation of all distances to the center of mass of the region.

7.2.5 η - Angluar Scan (ηAS)

In order to get a more complete and powerful representation of the shape, we present the following approach: From each SIFT point, and starting by the vector corresponding to the distance to the nearest border, an angular scan is performed so that every β degrees, the distance to the contour is computed and added as a new dimension in the feature vector. The scan has been performed in clockwise direction.

First, the vector to the nearest contour point is estimated. Then, an angular scan is performed clockwise by computing the distance every β degrees. Figure 19, shows the first vector in the left, the central image shows an angular scan

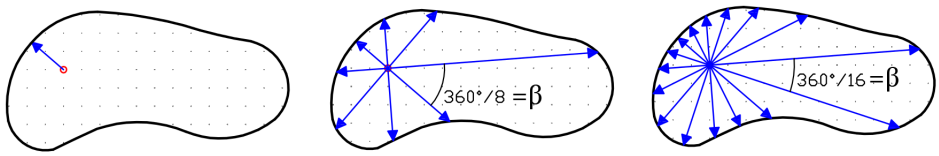


Figure 19: First vector, 8-Anguar Scan and 16-Angular Scan.

with 8 distances, and in the illustration at the right hand side shows an angular scan with 16 distances.

This design requires a parameter that will define the amount of angles that are considered. The precision of the partial quantization of the shape will be determined through this parameter.

$$\eta = 360/\beta$$

Rotation invariance is achieved for the angles in the same way as in the previous section(distance and angle to the nearest border). Rotation invariance for the sequence of estimated distances is achieved just by circular shifting the values of the distances until the biggest is located in the first position.

An important point in this approach in contrast to Shape Context [2] is that it does not go through holes in the shape. Once it finds a point in the contour, the distance is kept. This situation remembers to the human perception where a person can be looking around and somehow memorizing details in the shape of the surroundings and being finally able to recall the shape of the path.

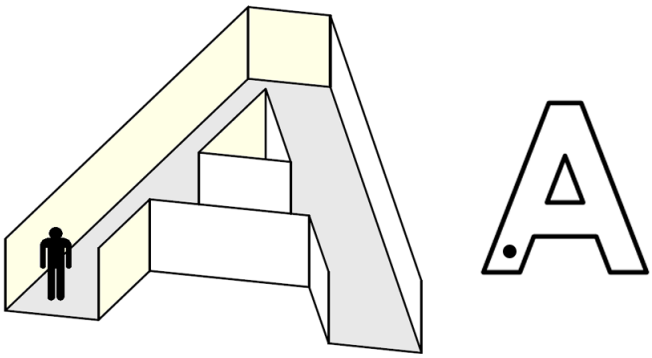


Figure 20: In the left image you can see the person going through a street, and not crossing over the holes, while in the right image the same street is plotted as a 2D representation.

7.2.6 Shape Context from a dense SIFT grid.(DSC)

The Shape Context [2]solution has been adapted to our framework of a dense grid of SIFT points inside the region to code the distance from each point to a subset of sampled points at the region's contour. All points in the grid are referred to the same set of contour points, as opposed to our **η-Angular Scan** descriptor, where each point is connected to the border points resulting from an angular scan around it. Points in the border are computed only once and common for all points, leading to a faster computation.

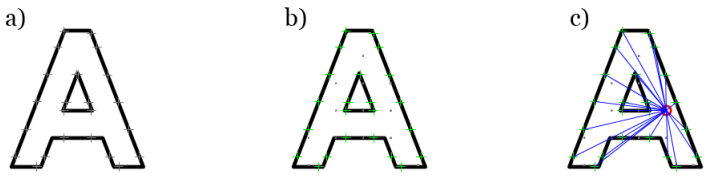


Figure 21: From each dense SIFT point of the grid (b), all quantized contour points (a) are reached and distances are kept (c).

In this case, note that the approach is also measuring the distances going through holes, as in the original Shape Context technique. An easy example can be the leg of a person. If a SIFT point falls into a leg, at some point there will be the computation of the distance from this point to one of the points of the other leg. In Figure 22 is a clear example where the distance is computed from one leg to the other one, crossing then, the limits of the region.

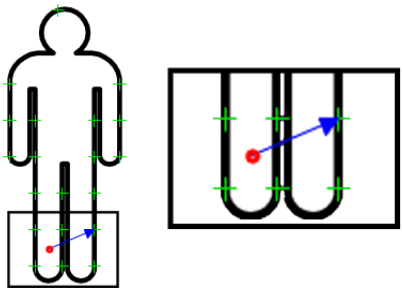


Figure 22: From a point inside one leg, the distance is computed by going outside the contours of the region.

7.2.7 Rotation Invariant Region Quantization (RIRQ)

One of the limitations of the Bag of Words approach is the loss of all information about the relative position of the points. Inspired by the Spatial Pyramid Matching technique[8], we have explored the possibility of adding a spatial partition of the region. The technique consists of a partition of the region in four quadrants, by choosing the major and minor axis direction as a cutting lines. Thus, it becomes rotation invariant. In each subregion, a region identifier will be set and added to all the points in that region.

Different encoding schemes are possible:

- **[1,2,3,4]**: It clearly identifies items, but the euclidean distance between 1 and 4 and between 3 and 4 is different. Thus, for the same distance in the 2D plane, the euclidean distance in the feature space is giving a different result, and this is going to affect to the k-means clustering and thus to the creation of the visual vocabulary.
- **[0001, 0010, 0100, 1000]**: In this case, both problems mentioned are solved. The euclidean distance between the values is always the same no matter the quadrant being analyzed.

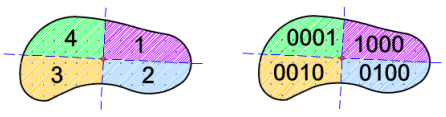


Figure 23: Different identifiers are set in the rotation invariant quantization

There is still a remaining question: How to decide which identifier corresponds to each quadrant.

To solve this issue, two solutions have been considered.

1. Sorting the regions by area.
2. Sorting the regions clockwise, starting by the partition with the biggest area.

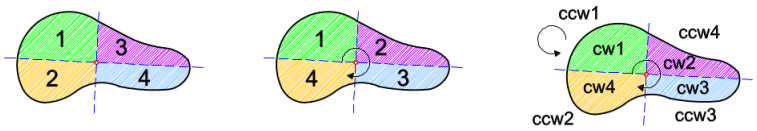


Figure 24: In the left image, the sorting of the subregions is done by area whereas in the centred image, the biggest area subregion is taken and from that, the other regions are identified clockwise. In the right image, the same approach as before is done, but not only with clockwise identification, also counterclockwise.

Flip invariance is also considered here. If nothing else is done, sorting by area is flip invariant, however, clockwise sorting is not. One way to get it would be to set the identifier to the subregions clockwise and counter-clockwise. Then, for a given point, compute the correct quadrants, so that there are two features. If a sorting over the features is performed, flip invariance is achieved.

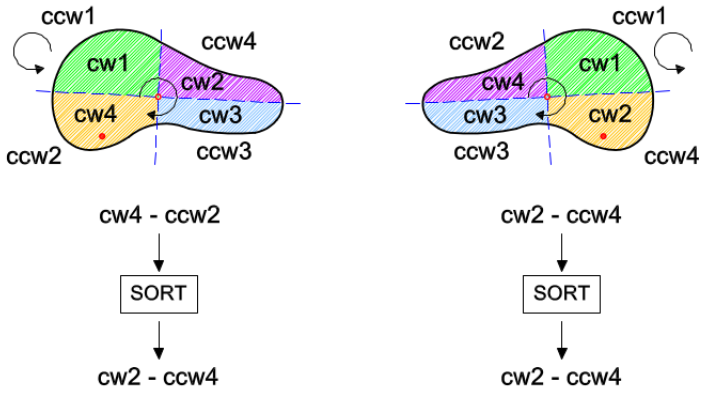


Figure 25: Example of flip invariance. First of all, from each point the right subregion identifiers are extracted both in clockwise and counterclockwise directions. Then, sorting is done achieving then, flip invariance.

7.3 Fusion before/after descriptors quantization

The proposed techniques are either point-based or region-based. Thus, the combination/integration in the BoW pipeline is different.

The different solutions for shape enrichment presented in [2, 6] can be combined with the SIFT-based descriptors at different moments. Here, two possibilities have been explored: Fusing them before the quantization of the descriptors in the vocabulary or after. Later on, they could be combined as well.

7.3.1 Fusion before descriptors quantization(eSIFT)

Once the SIFT descriptor is computed, with its 128-dimension descriptor, extra features are added to it in this step [11, 6], so that the final descriptors become 128+shape features dimensions long.

Figure 26: Adding extra features to the SIFT descriptor, an enriched SIFT descriptor.

In the literature, it is also referred as enriched SIFT [6]. There has been an improvement of performance by using this technique. In [11], a spatial pyramid is emulated by just adding a spatial reference(x and y, or θ and distance) to the end of the SIFT descriptor. [11]
Note that as descriptors are then quantized, the size of the histograms is not modified, and thus, the vocabulary size of the standard Bag of Words remains the same. It is expected that the correlation between the different features will be better captured by increasing the vocabulary size together with the enrichment of the descriptor.

By fusing all these new features, what is not yet defined is their dynamic range which will be different. Thus, a normalization step is required.

7.3.1.1 Normalization

At the descriptor level, some normalizations are possible:

- L^2 norm $p' = \frac{p}{|p|}$
- Power normalization[6]: $sign(p) \mid p \mid^2$
- Weighting spatial features
- Offset in spatial features to give them more relevance.

Improvements have been achieved in [10] by adding the x and y coordinates and normalizing SIFT features so that the sum of them is equal to 1.

7.3.2 Fusion after descriptors quantization(BoW+S)

This section illustrates the idea of increasing the dimensionality of the standard Bag of Words approach.

By using some of the shape features mentioned in the **Design** section, it is possible to create different shape histograms. The main idea of this is to explore the concatenation of the standard Bag of Words with other histograms that provide exclusively shape information.

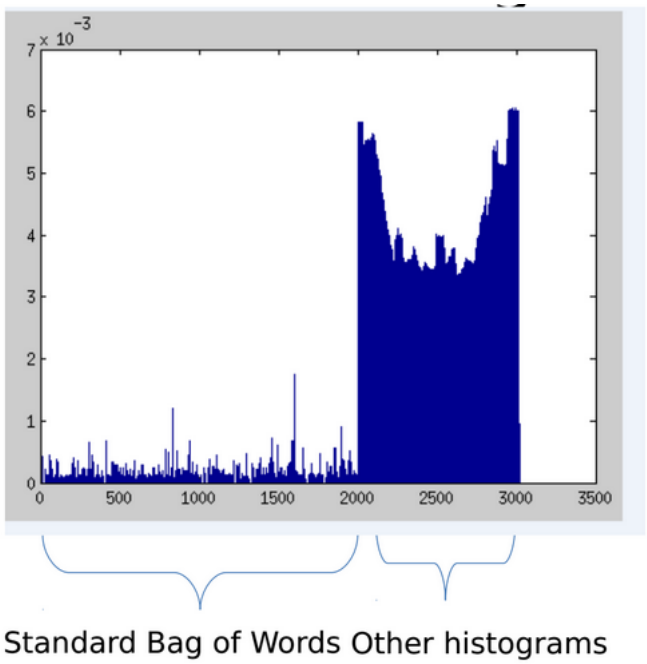


Figure 27: SIFT histograms(Texture information) + Spatial Histograms(Shape information)

7.3.2.1 Number of bins of the spatial histogram

An easy way to think about this point is to imagine the **distance to the nearest border(DNB)** approach which is based on the computation of a distance. In the BoW+S architecture, there is an array which accumulates all these distances. With this distances, a histogram is created. An important design parameter in the spatial histogram is the number of bins which will divide the size of the histogram and thus the shape vocabulary as $size = binsxshapefeatures$.

7.3.2.2 Adding more than one spatial histogram

In some of the approaches before explained, more than one feature is obtained. By taking this into account, is difficult to create a histogram for them as their meaning is different. In the case of a single feature, a histogram could be easily

created by accumulating all values of the specific feature. Extended to more than one feature, the treatment is the same, the only difference is that more than one spatial histogram is created, and then all of them concatenated.

This makes special sense in approaches like η-Angular Scan where each one of the angles is the relative angle to the vector that defines the starting point. The number of spatial histograms would be the same as angles, η. In this case, the final histogram is bigger than the proposed by the standard Bag Of Words.

7.3.2.3 Increase of relevance

Due to the fact that the size of the spatial vocabulary can be easily expanded, the relevance of the shape information becomes stronger than in the standard Bag of Words. An analysis of the relative size of the spatial histogram has been made and it is shown in the **Evaluation** chapter.

7.3.2.4 Normalization

If at histogram level, some bin is increased, then the importance of that part of the histogram is going to be more relevant than others. Therefore, normalization of the histogram plays an important role here.The following normalizations have been considered.

- Conversion to a probability density function.
- L^2 norm. $p' = \frac{p}{|p|}$
- log(Bag of Words): This normalization removes peaks produced by repetitive textures like the sky or tiles for example.
- Weighting spatial histograms.
- Offset in spatial features to give them more relevance.

8 Development

8.1 The VL_Feat framework

The application has been developed by using the Matlab VL_Feat Framework[1]. This framework includes a lot of useful functions such as SIFT, Dense SIFT, Multiscale PHOW SIFT, K-means, etc.

As a starting point, the Basic Object Recognition Application [2] has been used. It contains:

- PHOW features (dense multi-scale SIFT descriptors)
- Elkan k-means for fast visual word dictionary construction
- Spatial histograms as image descriptors
- A homogeneous kernel map to transform a χ^2 support vector machine (SVM) into a linear one. SVM classifiers

8.1.1 Structure of the application

The application is divided in several parts following the Bag of Words [7] pipeline with a χ^2 kernel map and a linear SVM classifier. It runs by default over Caltech-101 database, providing the possibility of computing the whole problem, or a reduced problem of 5 categories instead of 101. However, it can be easily customized.

1. Separate the training and test images.
2. Get SIFT descriptors.
3. Quantize the descriptors with the K-means clustering algorithm. Save the vocabulary
4. Compute Bags of Words by getting descriptors and quantize them into the closest descriptor in the vocabulary.
5. Run the χ^2 kernel map over Bags of Words
6. Train a linear SVM.
7. Test the data.
8. The algorithm reports a confusion matrix which is a NxN categories matrix, showing at which category the test images have been classified. It also reports the score matrix from the SVM, and the final performance value, the accuracy(%).

[1] http://www.vlfeat.org/
[2] http://www.vlfeat.org/applications/apps.html

8.1.2 Getting new descriptors and histograms

There are two major functions that encapsulate the novel features that we have proposed: The extraction of new descriptors and the creation of spatial histograms.

Firstly, the new descriptors are generated by using a single function that encapsulates all of them. This function returns several things:

- **frames**: In positions 1 and 2 are the x and y coordinates, while in 3 and 4 there are the scale and the orientation of the points.
- **descriptors**: They describe the neighbourhood of the points given by the frames. Additional features will be added to them.
- **Accumulation array**: This array is optional and gives an accumulation of the spatial features for the further creation of a shape-based histogram.

The convention of the vl_feat framework of returning frames and descriptors in all point-based descriptor functions has been followed as well in our own approaches.

9 Evaluation and Results

The main goal of this section is to answer to the following questions:

1. What is the best configuration in terms of accuracy?
2. What is the impact of the codebook size in the performance?
3. What is the impact of using object candidates with respect to ground truth masks?
4. What is the impact of the context in the Caltech-101 dataset?

An inherent problem in coding the shape is that the segmentation must be perfect. For the sake of the study in most of the cases we are going to suppose perfect segmentation and we will work directly with the ground truth of the data. However, real segmentation results are going to be provided as well.

9.1 Dataset Caltech-101

Caltech-101 [28] is a collection of pictures of objects belonging to 101 categories and the background category, making a total of 102 categories. There are about 40 to 800 images per category, being the average around 50 images per category. The dataset was collected in September 2003 by Fei-Fei Li, Marco Andreetto, and Marc 'Aurelio Ranzato. The size of each image is roughly 300 x 200 pixels.

Generally, only one single object appears in every image usually covering a big part of the area of the image. Another feature to take into account is that objects are sometimes not completely included in the image. This means that sometimes only a part of the shape will appear in the image. Rotation and flip problems are generally not appearing.

This dataset is a reference dataset used by the scientific community to test their algorithms.

9.2 Data partitions

For all categories, 30 of the images per category are chosen to belong to the training set and from 30 to 50 (depending on the number of images in that category) images per category are used as test images.

9.3 Metrics

In this section, the basic metrics used in order to evaluate the performance of all algorithms are explained:

- **Confusion matrix:** The confusion matrix is a NxN matrix where N is the number of categories. It counts the prediction results in a way that correct classifications are located in the diagonal of the matrix.

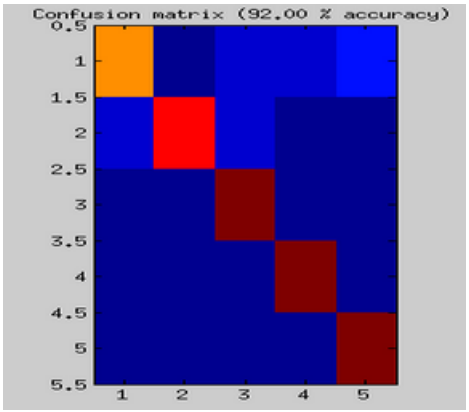


Figure 28: An example of confusion matrix in a reduced subset of 5 categories.

- **Accuracy:** This parameter measures the degree of closeness (in %) of the classification to the true value. Accuracy is calculated as follows:

$$Accuracy(\%) = 100\frac{TP+TN}{TP+FN+FP+TN}$$

Where TP(True positive) refers to positive samples that have been predicted good, FP(False positive) refers to positive samples that have been misclassified, TN (True negative) are negative samples which have been well predicted and FN(False negative) are negative samples which have been misclassified as well.

Accuracy(%) will be used to determine which approach is better.

9.4 Evaluation

This section shows the evaluation of all algorithms from the **design** section. Different vocabulary sizes have been tested (256, 512, 1024, 2048, 4096 and 8192) in order to see the variations when the size of the vocabulary increases.

9.4.1 Enrichment before quantization(enriched SIFT)

In this section, we show the results of the whole sets of experiments by trying to enrich the SIFT descriptor. All algorithms are referred by its acronym like in the **design** chapter.

Vocabulary	SIFT	eSIFT	DNB	LDNB	DANB	8AS	32AS	128AS	DSC	RIRQ	DC	SC
256	15,9477	18,7908	16,2418	16,6667	13,5948	19,8693	23,5948	41,3399	22,1242	16,7647	16,3072	15,6209
512	16,2745	20,9150	17,8758	18,5948	13,7255	21,6667	26,3072	41,8301	22,7124	17,4183	17,6144	17,1569
1024	17,8105	21,2745	18,1046	17,3856	15,5229	24,2157	29,0196	**42,1569**	25,13	17,6144	19,1176	17,9412
2048	17,8105	22,4510	18,4967	17,8105	15,5229	23,4967	30,0327	41,3399	27,15	18,2026	19,7386	18,5948
4096	18,5621	22,4510	17,7778	18,3007	15,2614	24,8693	31,2745	41,3399	27,9085	18,4641	19,5752	18,4314
8192	18,2026	22,5490	18,9542	18,3987	14,0196	25,7190	31,6993	40,915	26,7647	18,0300	18,5948	18,5294

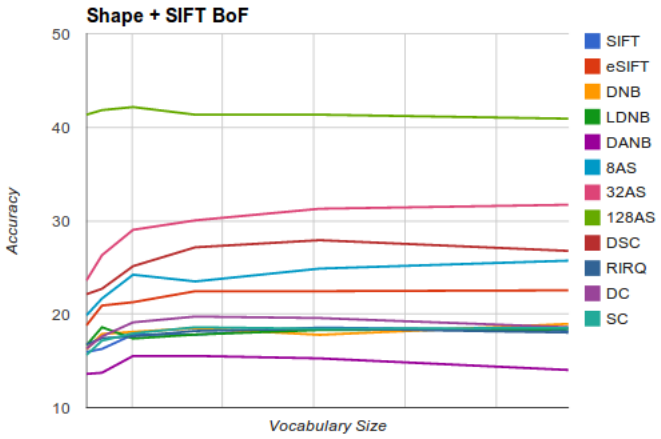


Figure 29: Result of the different enriched SIFT approaches.

The approach **128-Angular Scan (128AS)** has obtanied the best performance of **42,1569%** of accuracy. The improvement with respect to the standard dense SIFT is of about a **26%**. The second place is for the **32-Angular Scan (32AS)** with around **31%** of accuracy.

Small vocabulary sizes seem to show a tendency of increasing the performance but after reaching the point around 1024, the vocabulary size does not change much.

9.4.2 Enrichment after quantization(BoW-S)

This section includes all results from the enrichment of the standard Bag of Words by a shape-based histogram.

Vocabulary	SIFT	SIFT(Context)	DNB	LDNB	DANB	8AS	32AS	128AS	DSC	DC
256	15,98	29,8693	18,4641	16,5033	18,9542	29,1503	37,1242	41,2092	20,4248	20,4248
512	16,5359	32,0588	20,098	17,0261	19,7712	29,0196	37,549	41,83	20,8824	20,8824
1024	17,7778	31,9608	20,368	18,3333	21,5686	28,8562	38,2036	**42,1569**	21,1765	215686
2048	17,7451	33,6601	19,902	18,3007	20,5882	30,7516	38,366	41,3399	21,6667	20,5882
4096	18,4641	34,3791	20	19,183	20,1961	30,3595	**39,1503**	413399	23,1699	21,1961
8192	18,0392	34,7712	20,6209	18,4314	21,2418	30,1634	38,3007	40,915	22,4837	21,2418

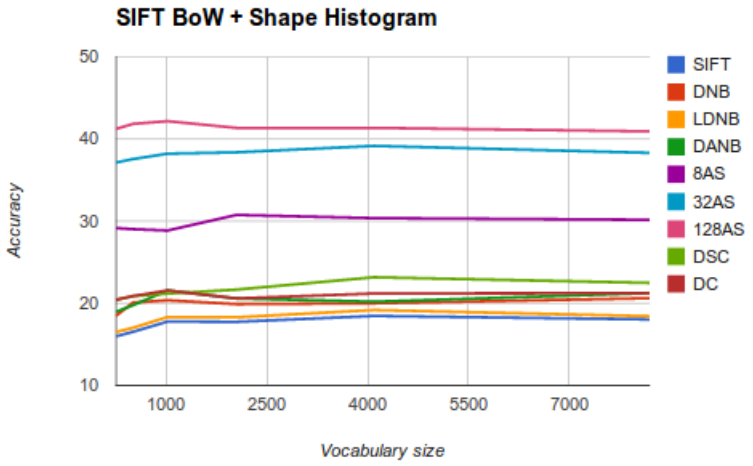


Figure 30: Result of the different BoW+S approaches.

In this approach, the best performance is also for the **128-Angular Scan** with a value of **42,1569%** of accuracy. However, the **32-Angular Scan (32AS)** is very close by achieving a **39,1503%** of accuracy. In this case, the difference between both is not as big as in the case of the enriched SIFT.

9.4.3 Comparison between enrichment before and after quantization

To clarify all the obtained results, a comparison between the enrichment before and after the quantization has been made. The Figure 31 illustrates the comparison between the best algorithms in both architectures.

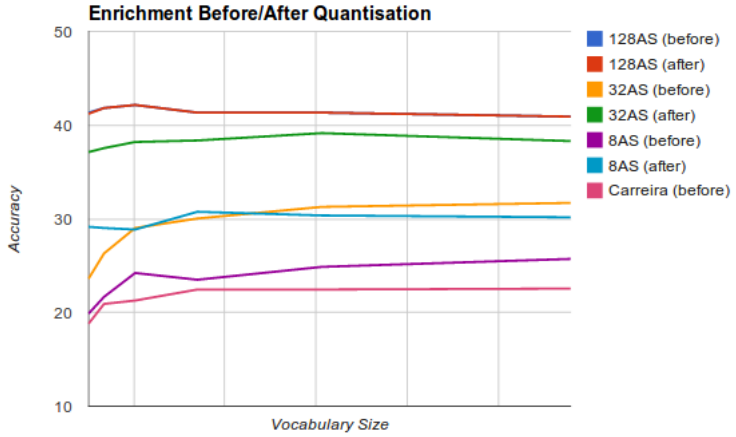


Figure 31: Comparison between the two different architectures studied. Note that the 128AS after and before are completely overlapped.

9.4.4 Influence of the number of bins

A study of the influence of the bins in the spatial histogram has been done in order to see the optimal number of bins. The analysis has been provided for one of the best approaches shown in the study (32AS) as there is not much difference between the best one (128AS) and the computation time is affordable. All results given in the study are computed with 8 bins per spatial histogram.

bins/histogram	32AS
1	17
2	28
4	35.817
8	**38**
16	34
32	29
64	24
128	19.902

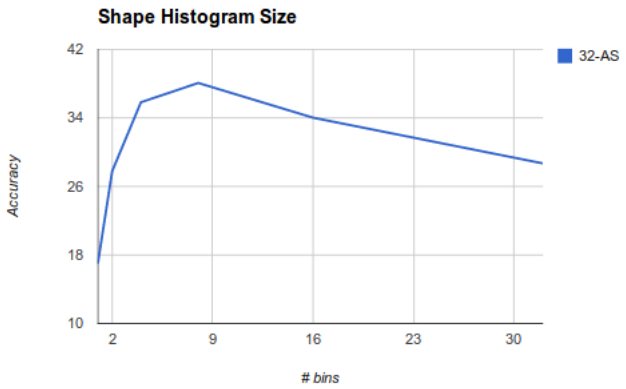


Figure 32: Effect of increasing the number of bits in the spatial histogram.

The number of bits shows a maximum around 8 bins per histogram while shows a decrease of performance when decreasing or increasing the number of bins for each histogram.

9.4.5 Comparison between ground truth masks and real object candidates

Working with ground truth segmentation provides results which do not correspond with most real applications, where no ground truth mask is available for the object. However, experimenting with them is convenient to see how much gain is possible to obtain by coding the shape. A comparison between ground truth and object candidates segmentation has been done to examine the decrease of performance by using real object candidates MCG segmentation.

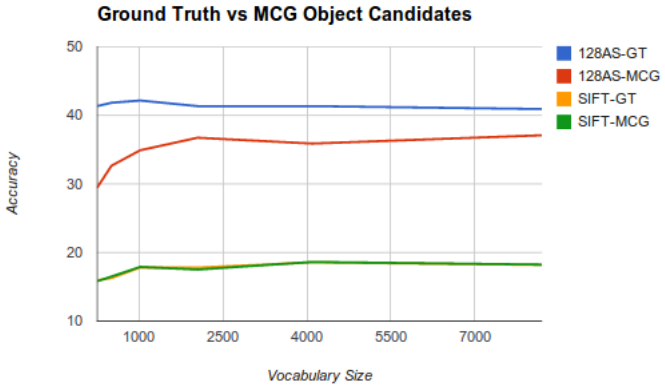


Figure 33: Comparison between ground truth and object candidates segmentation.

In Figure 33, we can see how the object candidates segmentation (MCG) is quite optimal when using SIFT features alone. However, when including shape features, for small vocabulary sizes, the difference of accuracy between both the ground truth and object segmentations increases. For big vocabulary sizes, both curves remain approximately constant.

9.4.6 Influence of the context

Due that in our algorithms, points outside the current region (known as context) have been neglected, we have decided to include an analysis of the context as well.

Vocabulary	Object (BoW + S, 128AS)	Object + Context (Baseline SIFT SPM)	Object (baseline SIFT BoW)	Object (BoW+S, 128AS-MCG)
256	41,3399	29,8693	15,9477	29,4444
512	41,8301	32,0588	16,2745	32,6471
1024	42,1569	31,9608	17,8105	34,9020
2048	41,3399	33,6601	17,8105	36,7320
4096	41,3399	34,3791	18,5621	35,8824
8192	40,915	34,7712	18,2026	37,0915

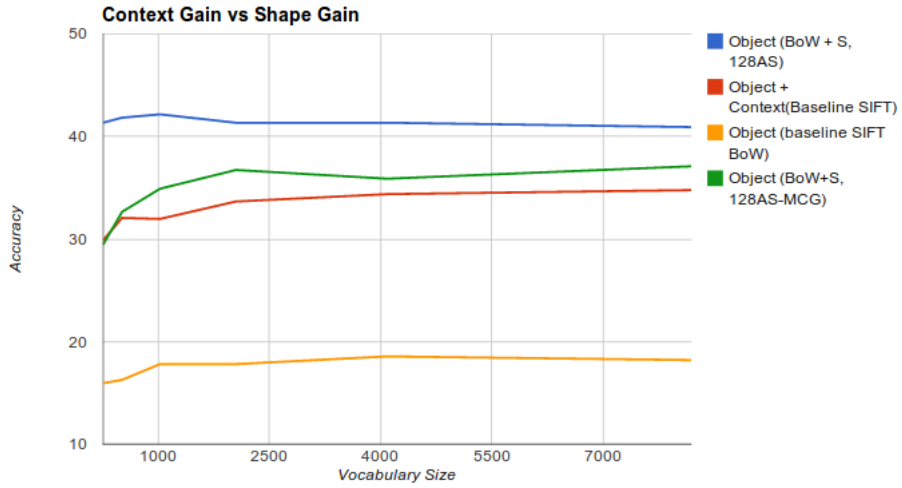


Figure 34: Analysis of the relevance of the context.

Figure 34 shows that by using the context information in the dataset Caltech-101 there is an increase of performance. However, intuition says if that the dataset were different, this increase of performance could result in even a decrease of performance.

10 Conclusions

This chapter aims to summarize the main conclusions of the whole study. We can conclude that:

Over all performed experiments with shape-based techniques in addition to standard SIFT descriptors, results have always shown an improvement in accuracy. Thus, the combination of dense point sampling and object candidates algorithm in order to codify the shape can work effectively in conjunction.

When the size of the standard BoW is low, the BoW+S pipeline gives much better results (around a 10% of accuracy improvement). However, when this size starts to grow, the enriched SIFT architecture as well as the BoW+S architecture tend to remain approximately constant. Nevertheless, BoW+S has shown better results in most of the proposed configurations.

The algorithm η-Angular Scan has outperformed over all the other experiments with its version 128-AS in an increase of around 31% of difference with respect to the SIFT baseline. The results of the two different architectures(enriched SIFT and BoW+S) proposed converge to the same results when the number of shape features added tend to high values.

Context has been proved to help in the Caltech-101 database. However, better performance has been achieved by codifying the shape rather than by looking at the context outside the region.

Shape-based features are sensitive to segmentation errors. Our experiments indicate that in a real object candidates segmentation (MCG) there is a decrease of performance around 5%.

However, in low sizes of the Bag of Words result in lower performances than higher sizes. Thus, the approaches are more robust to segmentation errors as the size of the visual vocabulary grows.

11 Future work

Distance to the nearest border(DNB)

In the figure 15, the distance to the nearest border from all points has been computed. Despite this is out of the scope of this project, it is possible to identify that each point in the border has a different density of vectors coming to it. It could be telling us the amount of bending that the location has.
In the same figure, it is possible to see a more accurate version of the skeleton of the region as lines inferred by the less dense area, and it would be interesting to study how to codify this version.

Rotation invariance

Most of the features added in this project are thought to be rotation invariant. However, the SIFT version used, dense SIFT is not rotation invariant, and this is why it is so fast.

An interesting study would be to try to make the dense SIFT rotation invariant to be able to let all the entirety of the approach be rotation invariant. By using DAISY descriptors instead of SIFT descriptors the approach would already have invariance.

Textureless object recognition

Due to the improvement in object recognition by using shape, it can improve results in textureless object recognition as well. It would be interesting to know how new features behave in front of a cartoon dataset or a textureless dataset like in [9].

12 Bibliography

References

[1] Belongie, S., Malik, J., Puzicha, J. (2002). Shape matching and object recognition using shape contexts. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(4), 509-522.

[2] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. PAMI, 27(11), 2005.

[3] M. Bober, "MPEG-7 Visual Shape Descriptors," IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. xref–x, June 2001.

[4] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110.

[5] Tola, E., Lepetit, V., & Fua, P. (2010). Daisy: An efficient dense descriptor applied to wide-baseline stereo. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 32(5), 815-830.

[6] Carreira, J., Caseiro, R., Batista, J., & Sminchisescu, C. (2012). Semantic segmentation with second-order pooling. In Computer Vision–ECCV 2012 (pp. 430-443). Springer Berlin Heidelberg.

[7] Sivic, J., & Zisserman, A. (2006). Video Google: Efficient visual search of videos. In Toward Category-Level Object Recognition (pp. 127-144). Springer Berlin Heidelberg.

[8] Lazebnik, S., Schmid, C., & Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on (Vol. 2, pp. 2169-2178). IEEE.

[9] Arandjelovic, R., & Zisserman, A. (2011, November). Smooth object retrieval using a bag of boundaries. In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 375-382). IEEE.

[10] Koniusz, P., & Mikolajczyk, K. (2011, September). Spatial coordinate coding to reduce histogram representations, dominant angle and colour pyramid match. In Image Processing (ICIP), 2011 18th IEEE International Conference on (pp. 661-664). IEEE.

[11] René Grzeszick, Leonard Rothacker, and Gernot A. Fink, "Bag-of-features representations using spatial visual vocabularies for object classification," in IEEE Intl. Conf. on Image Processing, Melbourne, Australia, 2013

[12] Ling, H., & Jacobs, D. W. (2007). Shape classification using the inner-distance. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 29(2), 286-299.

[13] Carreira, J., & Sminchisescu, C. (2010, June). Constrained parametric min-cuts for automatic object segmentation. In Computer Vision and

Pattern Recognition (CVPR), 2010 IEEE Conference on (pp. 3241-3248). IEEE.

[14] Boykov, Y. Y., Jolly, M. P. (2001). Interactive graph cuts for optimal boundary region segmentation of objects in ND images. In Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on (Vol. 1, pp. 105-112). IEEE.

[15] Zeppelzauer, M. (2013). Automated detection of elephants in wildlife video. EURASIP Journal on Image and Video Processing, 2013(1), 1-23.

[16] Ventura, C. (2013, October). Visual object analysis using regions and interest points. In Proceedings of the 21st ACM international conference on Multimedia (pp. 1075-1078). ACM.

[17] https://imatge.upc.edu/web/publications/bundling-interest-points-object-classification

[18] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/shape/sds/

[19] http://www.cs.berkeley.edu/ rbg/girshick2014rcnn

[20] Arbeláez, P., Pont-Tuset, J., Barron, J. T., Marques, F., Malik, J. (2014). Multiscale Combinatorial Grouping. CVPR.

[21] http://www.wiley.com/WileyCDA/WileyTitle/productCd-0471486787.html

[22] http://upcommons.upc.edu/handle/2099.1/9453

[23] http://link.springer.com/article/10.1007/s00530-002-0075-y

[24] David G Lowe, Distinctive image features from scale-invariant keypoints, International journal of computer vision 60 (2004), no. 2, 91–110.

[25] P. R. Beaudet, Rotationally invariant image operators, Proceedings of the 4th International Joint Conference on Pattern Recognition, 1978.

[26] Bay, H., Ess, A., Tuytelaars, T., Van Gool, L. (2008). Speeded-up robust features (SURF). Computer vision and image understanding, 110(3), 346-359.

[27] Navneet Dalal and Bill Triggs, Histograms of oriented gradients for human detection, Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, IEEE, 2005, pp. 886–893.

[28] Griffin, G., Holub, A., Perona, P. (2007). Caltech-256 object category dataset.