GESTALT-INSPIRED FEATURES EXTRACTION FOR OBJECT CATEGORY RECOGNITION

Patrycia Klavdianos, Alamin Mansouri, Fabrice Meriaudeau

GESTALT-INSPIRED FEATURES EXTRACTION FOR OBJECT CATEGORY RECOGNITION

Patrycia Klavdianos[⋆] *Alamin Mansouri*[†] *Fabrice Meriaudeau*[†]

[⋆] Queen Mary University of London
Department of Electronic Engineering (MMV Group)
Mile End Road, London E1 4NS, UK
[†] Université de Bourgogne
Laboratoire LE2I
BP 47870, 21078 DIJON CEDEX, France

ABSTRACT

We propose a methodology inspired by Gestalt laws to extract and combine features and we test it on the object category recognition problem. Gestalt is a psycho-visual theory of *Perceptual Organization* that aims to explain how visual information is organized by our brain. We interpreted its laws of homogeneity and continuation in link with shape and color to devise new features beyond the classical proximity and similarity laws. The shape of the object is analyzed based on its skeleton (good continuation) and as a measure of homogeneity, we propose self-similarity enclosed within shape computed at super-pixel level. Furthermore, we propose a framework to combine these features in different ways and we test it on Caltech 101 database. The results are good and show that such an approach improves objectively the efficiency in the task of object category recognition.

Index Terms— Region Self-Similarity, object category recognition, Gestalt, Semantic Grouping

1. INTRODUCTION

The *Gestalt* is a psycho-visual theory that was proposed in 1935 by Max Wertheimer, Wolfgan Kohler and Kurt Koffka in order to explain human visual perception through a series of mechanisms known as *Gestalt Laws* [1]. According to the gestaltists, visual perception obeys to some kind of perceptual organization from which the whole of a scene is not exactly the sum of its parts. Therefore, the scene and its parts play both important roles for a complete semantic understanding scenario. Hence, we believe that it is a kind of combination of these perceptual stimuli, from lower to higher levels of computation, that gives us ability to identify and categorize objects in the world. This motivates the current work which aims at a computational interpretation of the *Gestalt Theory of Perceptual Organization* laws, namely proximity, similarity, continuity of direction, closure or convexity, symmetry and appearance homogeneity.

However, the practical use of the Gestalt Laws is very challenging due to its higher level of subjectivity and the lack of a formal computational and/or mathematical representation. This fact has motivated researchers to provide models and test them in real-world applications.

Therefore, our main contribution is to derive a new model inline with the gestalt laws of perception and from which the input stimuli are organized in three semantic levels: inner, intermediate and global levels. We aim to represent the inner parts as well as the global appearance of the object as a combination of features and techniques from computer vision field. The features we propose are focused on the object itself (detected region) and in properties associated to its shape as well as the colors distribution within it. Regarding feature extraction and combination, our approach differs from others by providing: i) additional perceptual cues instead of only line segments, ii) utilization of line segments as a results of a skeletonization process, iii) feature representation is based on histograms as well as on their original vectors, iv) combining shape and appearance through the colors distribution and self similarity and v) application of two types of classification methods, based on AdaBoostM2 and through comparison of histograms using distance metric (*Quadratic-Chi - QC*).

The remainder of the paper is organized as follows: Section 2 describes in detail our method. Section 3 presents some results and Section 4 draws the conclusions and opportunities for future work.

2. PROPOSED METHODOLOGY

The global methodology we followed could be summarized in the following three steps: i) extract visual features based on the Gestalt laws, ii) combine and analyze the different features according to their physical and semantic meaning and iii) design different classifiers for object category recognition.

Regarding the skeletonization process we opt to use the technique introduced by Shen et al.[2]. And for the color distribution analysis, we segment the object internally using su-

perpixels based on SLIC *(Simple Linear Iterative Clustering)* technique developed by [3]. This solution is compliant with the Gestalt laws of similarity and proximity and gave us the possibility of exploring the inner parts of the objects.

Table 1 illustrates our approach and how we relate this work with the laws stated by the *Gestalt Theory of Perceptual Organization*.

Table 1. Our interpretation of the Gestalt Laws

Levels	Descriptors	Gestalt Laws
Global	Shape metrics	Proximity, similarity, continuity of direction
Global	Curvature analysis	Proximity, similarity, continuity of direction, closure
Local to Global	Self-similarity	Proximity, similarity, symmetry and color homogeneity
Local to Global	Shape decomposition	Proximity, similarity, continuity of direction
Local	Shape transform	Proximity, similarity
Local	Color-Location Transformation	Proximity, similarity and color homogeneity

2.1. Capturing the inner/local appearance

Our approach combine features build on skeleton, color distribution and shape analyses. Regarding the local appearance of the object, we used two types of descriptors: based on shape transform and based on Color-Location transformation.

2.1.1. Descriptors based on shape transform

The descriptors based on shape transform utilize Fourier features constructed from the coordinates (x_k, y_k) defining the object boundary. Let (x_k, y_k), $k = 0, 1, ..., N-1$ be the coordinates of N samples on the object boundary. For each pair (x_k, y_k) we define the complex variable as follows:

$$u_k = x_k + jy_k \tag{1}$$

For the N u_k points we obtain the DFT (Discrete Fourier Transform) fl:

$$fl = \sum_{k=0}^{N-1} u_k \exp(-j\frac{2\pi}{N}lk), l = 0, 1, ..., N-1 \tag{2}$$

2.1.2. Local Color distribution

This descriptor is based based on local color distribution and is calculated upon a transformation between (RGB) coordinates belonging to a *superpixel* and their relative spatial coordinates. If we can extract a feature inside each superpixel by relating its color coordinates with its relative spatial coordinates, the resulting feature must hold information about spatial distribution of colors in an invariant way. The invariance comes from the properties of the covariance of spatial coordinates when an affine transformation is applied to their corresponding color coordinates [4]. Moreover, since we used for each superpixel not the absolute spatial coordinates but the relative ones (considering the center as the origin), these relative coordinates are invariant to scale and orientation.

Considering that each position (x, y) in the superpixel region is also represented by a color triplet (R,G,B), it must exist a transformation from which we can map the relative coordinates of a pixel to the color and vice-versa. From this we can define the following equation:

$$Position_{(x,y)} = Color_{(R,G,B)}T1 \tag{3}$$

where $T1$ is the transformation which maps *Color* to *Position* of pixels within a superpixel structure. T1 can be calculated by means of least square solution and pseudo inverse approximation. After that, we apply $T1$ to Color in order to obtain:

$$\widehat{Position}_{(x,y)} = T1Color_{(R,G,B)} \tag{4}$$

Then, we compute a second transformation as follows:

$$\widehat{Position}_{(x,y)} = Position_{(x,y)}T2 \tag{5}$$

We can approximate the result of the previous equation by applying again *least-square*. Finally, we end up with $T2$ which holds the information about color distribution regarding its spatial location. $T2$ is a 2x2 matrix and we use this values to create our descriptor. Therefore, for N_b superpixels we will get 2 X Nb features.

2.2. Capturing the global appearance

For extracting information about the global appearance of the object, we used two types of descriptors: based on shape measurements and on curvature information.

2.2.1. Descriptors based on shape measurements

In this category one can include the classical shape analysis metrics such as: area, perimeter, eccentricity, solidity, irregularity, major and minor axis, centroid, elongation, circularity, etc.

2.2.2. Descriptors based on curvature information

Regarding the curvature, we use two descriptors. The first one is based on the chain code proposed by Freeman [5] which could be interpreted as compliant with the Gestalt law of proximity. In our implementation we used the 8-neighbor

connectivity model. We also guarantee invariance to the choice of the starting point and to rotation. The second curvature descriptor is constructed also from the chain code result (without applying the first difference). The chain code is grouped according to features that quantify the concave and convex external angles between adjacent edges at the corners of the polygon that is formed by the line segments when the boundary curve is scanned in the clockwise sense. We used the patterns for concave and convex features suggested in [6]. Fig. 1 gives an illustration of some shape measurements.

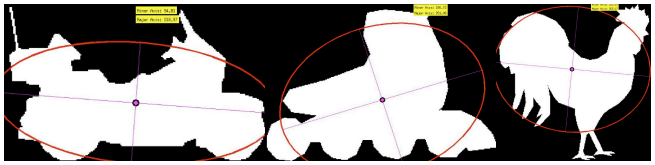

Fig. 1. Shape Measurements

2.3. From local to global appearance

We also provide two feature descriptors that aim to capture the object characteristics in an intermediate level of semantics that we called "local-to-global" stage. From this, we used the following descriptors: based on shape decomposition and on self-similarity.

2.3.1. Descriptors based on shape decomposition

The shape decomposition descriptor is based on the object skeleton. From the skeleton, we perform shape decomposition by detecting its end points and junctions (T, L and X junctions) and afterward by separating the skeleton in line segments. These lines segments are, thus, used to build features using histograms of the relative lengths and the angles between line segments like in [7]. However, we use only the segments resulted from the skeleton structure which holds itself physical meaning about the object formation. Such a process is illustrated in Fig. 2.

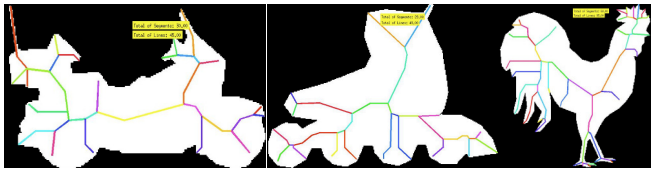

Fig. 2. Shape Decomposition

2.3.2. Self-similarity

Our intuition regarding self-similarity is that the object contains internal similar structures which contribute to define its overall appearance. Therefore, we devised a more compact self-similarity descriptor based on the work of Irani et al. [8]. In our self-similarity descriptor we used *superpixels* to obtain a nearly homogenous set of regions. For each region we compute the similarity against the others. The result of the computation is a self-similarity surface which is represented by a symmetric matrix of size NxN where N is the total number of *superpixels*. The resulting distance surface is, thus, calculated with *Sum of Squared Differences (SSD)* as follows:

$$SSD_q(x,y) = \sum (SP_k(x,y) - SP_l(x,y))^2 \qquad (6)$$

Where $SP_k(x,y)$ is a pixel inside the *superpixel* region taken as reference and $SP_l(x,y)$ is a pixel on other *superpixel* region taken for comparison. Our self-similarity metric compares each pixel inside a *superpixel* with the equivalent pixel in other *superpixel*. This approach suggests that the comparison between *superpixels* obeys to some well defined spatial arrangement. However, in reality we cannot guarantee that. Therefore, to overcome this problem we calculate the self-similarity for those pixels that can be compared in terms of their sizes. This means to say that we first check for the similarity in size and then proceed with the SSD computation. After some experiments, we realized that two *superpixels* could be considered similar in size if the total number of pixels of the smallest one is at least 70% of the total number of pixels on the other. In doing so, we pre-selected the *superpixels* which could effectively participate in the comparison.

Another problem to solve regarding self-similarity implementation involves the comparison itself: how to compare *superpixels* in different shapes, sizes and location. In order to do so, we defined three squared regions with the size of the biggest *superpixel* participating in the comparison. Each one of these regions will store the color values (red, green and blue) of the pixels inside the region. Then, we copy each equivalent pixel in the *superpixel* region to its correspondent channel in sequential order, meaning that the first pixel is copied to the first position in the squared region and so on. At the end, we have the two *superpixels* represented by squared regions with the same size. The unfilled areas in the three squared regions have its values defined by pixel symmetry technique. Our symmetry technique comes from the idea that *superpixels* tend to spread the same pattern along the region. Therefore, if we resize the *superpixels* or arrange them in a squared window, in order to complete the unfilled areas we could just repeat its same pattern. In order to fulfill the uncompleted areas, we repeated the color pattern of the adjacent regions. That way, we are not only totally compliant to the idea of proximity and similarity, but we also introduced a new approach of symmetry based on the inner pattern of a *superpixel* region.

Finally, the result of the SSD metric is, thus, normalized and transformed into a descriptor vector which represents the self-similarity for the entire object. The following equation describes the normalization step:

$$SSD_q(x,y)_{norm} = \exp\left(-\frac{SSD_q(x,y)}{max(SSD_q(x,y))}\right) \quad (7)$$

The self-similarity vector generated for different images can have different sizes since it depends on the total number of *superpixels*. Therefore, we provided two representations for this descriptor, the original and other one based on histogram. The histogram vector is applied in AdaBoost classifier while the original is used in the classifier based on the distance metric. This way, we can test the efficiency of self-similarity descriptor by using two different approaches. Fig. 3 gives an illustration of our self-similarity method.

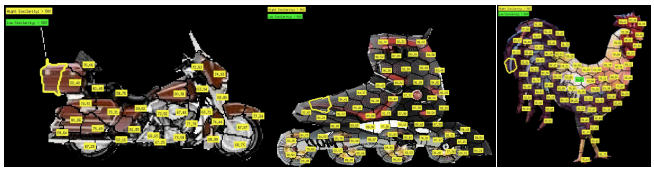

Fig. 3. Self-Similarity

3. EXPERIMENTS AND RESULTS

In order to evaluate the features and the method we propose, we developed a framework that receives images and their corresponding annotations files as inputs and generates the names of the object classes. As data, we used *Caltech 101* [9] because it is considered as one of the challenging databases with large interclass similarities and intraclass variations but also because of the availability of the ground truth and annotations.

The combination of features as well as the type of classifier considered in our testing phase are summarized in Fig. 4.

In a general way our approach is comparable to other methods. We obtained a total mean recognition rate per class of 41% considering 15 classes without any change on parameter. This score is better than the recent methods we compared with (Table 2).

Table 2. Classification rates for Caltech 101

Model	Performance (rate/class)
Our method	41%
Holub et al. (2007)	37%
Serre et al. (2005)	35%
Fei-Fei et al. (2007)	18%
SSD Baseline	18%

One can notice that our method performs better than Fei. Fei et al. [10], SSD Baseline, Serra et al. [11] and Holub et al. [12]. Our method obtains quite similar results to those

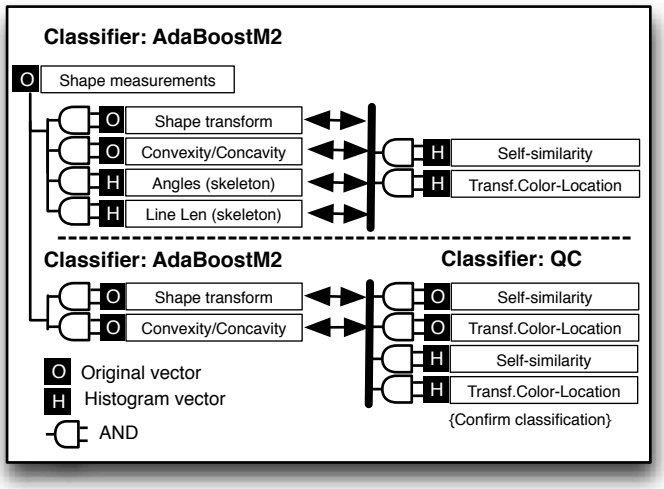

Fig. 4. Feature combination and classifiers

reported in [7]. However, regarding the results presented by these authors, we would like to mention that we did not get the scores claimed by the authors when we implemented ourselves their solution. The scores we obtained are much lower than those of the article when we keep the same thresholds for all classes. This may suggest that the authors are using different thresholds for each class which is like having a customized and well defined solution for every problem. In our case, we propose a unique and unparameterized solution for the general problem of object categorization.

The classes which performed better in our solution are motorbikes (89%), inline skate (85%) and rooster (81%). The classes showing poor performance are anchor (12%) and schooner (14%).

4. CONCLUSION

The main concern of this work was to extract visual information, combine them in lower and higher level semantic groups and test their efficiency for the purpose of object categorization. The results have shown that our method is effective and presented new ways of exploring object categorization by combining shape and appearance based on color distribution.

The most important finding in our investigation is associated to the perceptual meaning of the self-similarity descriptor. Our outcomes showed that not only our self-similarity method captures the topology of the objects, but translates perfectly the concept of symmetry among the object's parts.

Therefore, for future work we aim to improve our self-similarity method to include spatial context information so that we can represent the object according to its most salient parts. Other aspect to be explored is to consider complementary features associated to the skeleton structure and a more detailed investigation on different types of classifiers.

5. REFERENCES

[1] Stephen E. Palmer, *Vision science : photons to phenomenology*, MIT Press, Cambridge, Mass., 1999, Stephen E. Palmer.; "A Bradford book."; Bibliography: Includes bibliographical references (p. [737]-769) and indexes.

[2] Wei Shen, Xiang Bai, Rong Hu, Hongyuan Wang, and Longin Jan Latecki, "Skeleton growing and pruning with bending potential ratio," *Pattern Recognition*, vol. 44, no. 2, pp. 196–209, 2011.

[3] Radhakrishna Achanta, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Ssstrunk, "Slic superpixels ," 2010.

[4] Xiaohu Song, "Descripteurs couleur locaux invariants aux conditions dacquisition, phd thesis," 2010.

[5] "Freeman chain code (fcc).," in *Encyclopedia of Biometrics*, Stan Z. Li and Anil K. Jain, Eds., p. 592. Springer US, 2009.

[6] Sergios Theodoridis and Konstantinos Koutroumbas, *Pattern Recognition, Third Edition*, Academic Press, Inc., Orlando, FL, USA, 2006.

[7] Nishat Ahmad, Youngeun An, and Jong-An Park, "An intrinsic semantic framework for recognizing image objects," *Multimedia Tools Appl.*, vol. 57, no. 2, pp. 423–438, 2012.

[8] Eli Shechtman and Michal Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recognition 2007 (CVPR'07)*, June 2007.

[9] Li F. Fei, Rob Fergus, and Pietro Perona, "One-Shot Learning of Object Categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, 2006.

[10] Li Fei-Fei, Rob Fergus, and Pietro Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Comput. Vis. Image Underst.*, vol. 106, no. 1, pp. 59–70, Apr. 2007.

[11] T. Serre, L. Wolf, and T. Poggio, "Object recognition with features inspired by visual cortex," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, june 2005, vol. 2, pp. 994 – 1000 vol. 2.

[12] Alex D. Holub, Pietro Peron, and Max Welling, "Exploiting Unlabelled Data for Hybrid Object Classification," in *Neural Information Processing Systems*, 2005.