CVC Tech.Rep. #110 July, 2007

Blob detection and grouping for texture description and other applications

Anna Salvatella and Maria Vanrell

Computer Vision Center / Dept. Informàtica (UAB)
Edifici O - Campus UAB
08193 Bellaterra (Barcelona)

Submitted to the Computer Vision Center
on July, 2007

Contents

Blob detection and grouping for texture description and other applications

by

Anna Salvatella and Maria Vanrell

Abstract

The goal of this report is to present some work on blob detection and grouping developed in the context of a wider project of image content interpretation where the main topics are color, texture and its interaction.

In this work the problem of texture representation for natural textures is approached from a blob-like point of view and several applications derived from this approach are presented.

Firstly the assumption that natural images can be understood as sets of Gaussian blobs is presented. From this point, a blob detection algorithm based on scale-space representation of images is presented, yielding a blob decomposition of images.

A second step is based on the assumption that in most natural images a number of groups of similar arise. In order to find these blobs' groups a blob representation space with perceptual properties is defined. In such space, blobs that are perceived as belonging to a same group should be represented by close-by points. The grouping of similar blobs is performed by grouping the points representing them in this perceptual space by the N-cut algorithm.

Finally, the last section of the paper is devoted to showing several applications of this blob's groups image representation.

1 Introduction

The growing of digital image databases makes it necessary to have instruments which are able to automatically find images in large sets according to useful criteria. These tools should be able to list all the terms a user could possibly use to describe what is present in the image and also what is conveyed from it. Several steps have already been done towards this difficult goal [1]. Nevertheless, the semantic gap, defined as the distance between what is perceived from an image by a user and what is inferred from that image in a computational point of view, has not been completely bridged yet, and thus it is still limiting the results of these tools.

In this paper we focus on an image representation of texture which is based on how users perceive and describe textures, and thus which could be used to describe textures with automatic annotation tools in the frame of the MPEG-7 [2].

Although there is a lack of a standard definition for texture, we can say there is an agreement from several works [3, 4, 5] that a texture can be described by geometrical features related to the scale, the orientation and the regularity of the local features, called sometimes textons [6, 7]. The most successful attempts to describe textures have been based on the analysis of frequential information with large set of filters, followed by heuristic functions defined ad-hoc to infer the geometrical features. In this paper we propose to work just on those filters that allow to directly detect the image blobs and their geometrical features. To this end, we assume an image is a set of Gaussian blobs with its specific properties, as it is done by Lindeberg in [8], and close to the *Marr's primal sketch* [9].

According to the previous assumption a texture image can be described through the description of its blobs and their attributes. Blob attributes are shape (given by its width and its length), contrast, orientation (in case it is elongated) and location in the image. We will denote this representation as the *blob decomposition (BD)* of an image.

As it will be shown, such a representation is enough to characterize textures, but it is still far to be suitable for browsing or searching applications. To go further in this sense we propose to work on *blob groups* instead of just working on blobs. We will define a *blob group as a set of perceptually similar blobs*, i.e. blobs sharing similar geometric properties. We will denote this representation as the *blob group decomposition (BG)*.

In the attempt of bridging the semantic gap, using *BG* to build image descriptions can present interesting properties. Firstly, it is a *Non-Subjective description*, that is linked to geometric image properties which can be quantified, and not just numerical representations correlating with subjective judgements, such as *coarse* or *fine*. Secondly, it is a *Semantic description*, that can be obtained for each blob group, since linguistic terms describing the geometric properties shared by the blobs in this group are easily derived, i.e., *"dark 45^o-oriented blobs"*. Finally, the description is *hierarchical*, since the combination of blob groups may allow to derive linguistic terms on a higher level, i.e., a *"chequered pattern"* from two blob groups formed by elongated blobs with orthogonal orientations.

In section 2 we propose a complete method to automatically obtain the *BD*, based on the blob detection procedure of Lindeberg [8]. In section 3 we propose a uniform blob space where distances rely on similarity of blob attributes. This space is the basis for the blob grouping procedure presented in section 4, based on the NCuts clustering technique. Finally, examples and applications of both representations are given in section 5.

2 Blob detection in scale-space images

To decompose the image in its blobs we have focused on the use of differential operators in its scale-space representation as suggested in [8], where a blob detector which suits our purpose is defined. All the blobs of the image are supposed to be Gaussian-like and will be characterized by their width, w, their length, l, the orientation of the major axis, θ, and their contrast, c. The aspect ratio, ar, and the area, A, can be defined from these attributes as $ar = l/w$ and $A = l \cdot w$.

The blob detector is based on Laplacian of Gaussian filtering of the scale-space representation of the image defined by $L_\sigma(I) = I * G(\cdot;\sigma)$. In order to automatically detect the scale of the blobs, the *normalized* differential Laplacian of Gaussian operator, $\nabla^2_{norm} L_\sigma = \sigma^2 \nabla^2 L_\sigma$ is used, so that the centers of the Gaussian blobs are given by extremes over scale and space of $\nabla^2_{norm} L_\sigma$, and their scale is given by the scale at which the extremum has been achieved, denoted as s_{LoG}. The value of the extremum gives information about the contrast of the blob and its sign tells whether the blob is bright or dark.

In case the blob is not isotropic and thus has two different characteristic lengths σ_1 and σ_2, the maxima of $\nabla^2_{norm} L_\sigma$ is achieved at a scale proportional to $\sqrt{\sigma_1 \sigma_2}$, but no information about the values of these two characteristic lengths is given. Therefore, some other measure must be considered to obtain the shape of elongated blobs.

For this purpose, Lindeberg introduced a differential entity, the windowed second moment matrix (*WSMM*). This operator is meant to study the grey-level structure around a point q of the image, and is defined by

$$\mu_L(q) = E_q \begin{pmatrix} L_x^2 & L_x L_y \\ L_x L_y & L_y^2 \end{pmatrix} \tag{1}$$

where E_q denotes an averaging operator centered at $q = (x, y)^T \in \mathbb{R}^2$.

This operator has already been used in the definition of texture features [10] but as a local structure descriptor applied to all points of the image and not for blob shape detection. The information given by the *WSMM* is easily interpreted in terms of its eigenvectors and eigenvalues: since this matrix is positive semi-definite, the equation $(\xi - q)^T \mu_L(q)(\xi - q) = 1$ defines an ellipse. The orientation of the major semi-axis of the ellipse is given by the eigenvector corresponding to the smallest eigenvalue, and the lengths of the semi-axes are given by the square root of the inverse of the eigenvalues. Thus, the aspect ratio and the orientation of the blob are given by

$$ar = \sqrt{\tfrac{\lambda_2}{\lambda_1}} \quad \theta = \arctan(\tfrac{v_2}{u_2}) \tag{2}$$

where λ_1 and λ_2 are the eigenvalues of the *WSMM* in decreasing order and (u_2, v_2) are the coordinates of the eigenvector corresponding to λ_2.

The calculation of *WSMM* involves two scale measures, the *integration scale (s)*, the scale of the averaging operator, and the *derivation scale (t)*, the scale of L_σ or smoothing scale. The choice of these two scales can widely vary the results [11]. A complete scheme for the calculation of the suitable scales is proposed in [8], but the implementation of this method is quite costly, in other works by the same author several shortcuts have been proposed to deal with this problem in specific applications [12, 13]. In our work, the characteristic size of the blob is close to the value of s_{LoG}. Therefore, the integration scale is set to be proportional to this characteristic value: $s = \gamma_1 s_{LoG}$. The scheme proposed by Lindeberg to calculate the derivation scale is based on an anisotropy maximization criteria in order to compensate the shape distortion due to isotropic smoothing. Nevertheless, in our approach, which is meant to be able to deal with highly non-Gaussian blobs, another criteria is chosen, since we consider it is more suitable to work with the scale-space representation of the image corresponding to the scale of the blob. By doing so, the smoothing is performed at a scale which is a compromise between the conservation of the blobs shape and a smoothing that enables a more Gaussian profile of the blobs. Thus, the two scales are chosen to be proportional to s_{LoG} with $s = \gamma_1 s_{LoG}$ and $t = \gamma_2 s = \gamma_1 \gamma_2 s_{LoG}$.

At this point the area of the blob is still unknown, since the blobs size can not be deduced from *WSMM*. In previous work this has been solved by fixing the area of the blob proportional to s^2_{LoG}, but we propose to set s_{LoG} as the width of the blob instead. This assumption does not modify the results in case of isotropic blobs and it improves the results in case of elongated blobs, where it is the width of the blob which tunes better with the filter. Therefore, the blobs width and length are defined by $w = s_{LoG}$ and $l = ar \cdot s_{LoG}$. In figure 1 the results of applying this criteria are shown. As it can be seen, in case the blobs are highly anisotropic the assumption $w = s_{LoG}$ improves the results.

Given an image I, the detection and characterization of its blobs yields to its Blob Decomposition *BD*, which is represented by a $M \times 6$ matrix $\mathbf{B}$, whose rows are defined as

$$\mathbf{B}^I_i = (x, y, w, l, \theta, c) \tag{3}$$

where (x, y) denote the location of the blob in the image and c gives the contrast of the blob. The *BD* for a natural image is shown in figure 2, where blobs are superimposed on the lightened image.

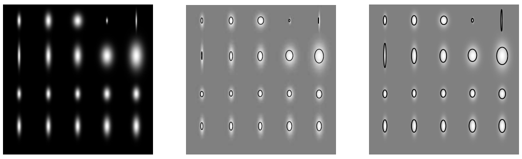


Figure 1: Results of blob detection and shape estimation for Gaussian blobs. The image is lightened and the blob's contours are superimposed. (a) Original image. (b) Area of the blob equal s_{LoG}^2 (c) Width of the blob equal to s_{LoG}.

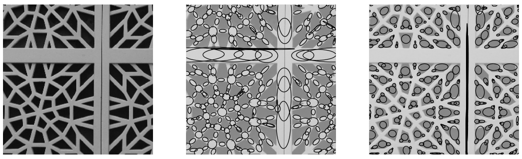


Figure 2: *BD* for a natural image (a) original image (b) bright blobs (c) dark blobs

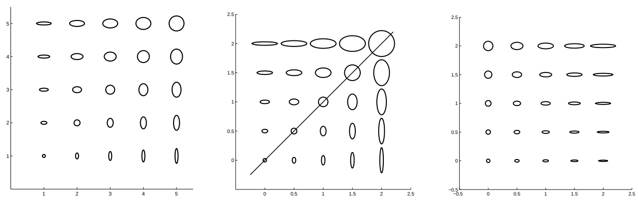
3 A uniform blob space

Once the Blob Decomposition of the image is obtained, we aim to group its blobs according to the similarity of their perceived attributes. We propose a uniform space, where the distance between two points is proportional to the difference between its perceived attributes, as uniform spaces are defined in color science [14]. To derive this blob space we will firstly define the two axes corresponding to shape. Afterwards, orientation and contrast will be introduced to yield a three-dimensional space.

The simplest way to represent blob shape is by directly associating the two shape attributes of the blobs, resulting in a length-width (l-w) space. In figure 3.(a) some blobs are represented according to these coordinates. As we are dealing with digital images, both attributes of the blobs start at a value of 1 pixel and therefore there is a shift between the origin of the coordinate system and the smaller possible blob. This problem can be solved by introducing a logarithmic scale so that the blobs with these initial values are at the origin of the axis, which suits better our purposes. The representation of blobs in logarithmic scale is shown in figure 3.(b).

At this point, let us notice that half of the blobs that are plotted are not valid, since by definition the width is the shortest of the two characteristic lengths of a blob, and blobs plotted in this plane are supposed to have the same orientation. Thus, blobs in each side of the axis defined by the isotropic blobs are redundant (see figure 3.b). To remove these blobs we propose to rotate the axes so that isotropic blobs lie in the vertical axis, and just leave those remaining in the first quadrant. This transform will link the new axes to $\log(ar)$ and $\log(A)$ features, as shown in figure 3.(c).

Once shape attributes have been associated to specific axes on the blob space, now we have to add the orientation and the contrast. Considering that isotropic blobs, i.e. non-oriented blobs, are situated just along the area axis, it seems natural to introduce the orientation of blobs as an azimuthal coordinate of



(a) (b) (c)

Figure 3: Blob spaces. (a) l-w space, (b) $\log(l)$-$\log(w)$, (c) $\log(ar)$-$log(A)$ space, proposed in this work

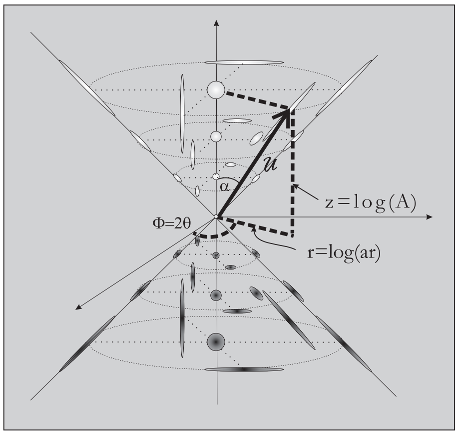


Figure 4: Blob space in cylindrical coordinates

a cylindrical space, that organize the orientation differences along a ring centered on the area axis, this will be obtained by doubling the angle. Finally, in order to represent both bright and dark blobs in the space but separately, we propose to use the sign of the contrast on the vertical axis, with bright blobs corresponding to positive values of the contrast.

Hence, the radial, azimuthal and vertical coordinates, (r, z, ϕ), of this blob space will be related to aspect ratio, orientation and area of the blob, respectively. A graphical representation of this space can be seen in figure 4.

According to all the considerations we denote as $\mathcal{U}$ the non-linear function transformation that takes to the new space, $\mathcal{U} : \mathbb{R}^4(w, l, \theta, c) \longrightarrow \mathbb{R}^3(r, z, \phi)$, and given by

$$\mathcal{U}(w, l, \theta, c) = (log(ar), sign(c) \cdot log(A), 2\theta) \tag{4}$$

In figure 4 we show the location of several groups of blobs in the proposed space. The definition of the space implies all blobs are represented in a cone delimited by an angle $\alpha = \pi/4$. The cones are drawn in figure 4 as the shaded volume where points representing blobs are contained.

Although this space seems to be obtained from computational considerations only, there are perceptual reasons that show it can be considered as a uniform blob space. It has been shown [15] that the evaluation of the area in the human visual system is performed by a specific mechanism, and not by the multiplication of the dimensions of the ellipse. Equally, it is also demonstrated [16, 15] there is some mechanism in charge of evaluating the aspect ratio of 2-D shapes. Furthermore, the independence of evaluation of the area and aspect ratio has also been demonstrated in [17].

The similarity property of this uniform space will provide meaningful interpretations in a perceptual sense. In figure 5 we show several interesting groups of blobs that could be described by parametric functions.

Blobs sharing the same shape and area but different orientations will form a group laying in a ring or a part of a ring (if the orientation is in an interval) around the vertical axis. The distance due to differences in orientation increases with the aspect ratio of the blob, as does the radius of the ring. In figure 5 points corresponding to several blobs with same shape but different orientations are plotted forming an arc.

Isotropic blobs lie in the vertical axis of the space, making groups of isotropic blobs with similar area become quite clear, as shown in figure 5.

Blobs having similar area and orientation but variable aspect ratio lie in a line perpendicular to the vertical axis, as the area is constant.

Finally, blobs having the same width and different length lie in lines forming an angle of $\alpha = \pi/4$ with the vertical axis. This group of blobs often appear when the calculation of the *WSMM* underestimates the anisotropy of the blob. Equally, blobs with the same length but with different width lie in lines forming an angle of $\alpha = -\pi/4$ with the vertical axis. Both cases are illustrated in figure 5.

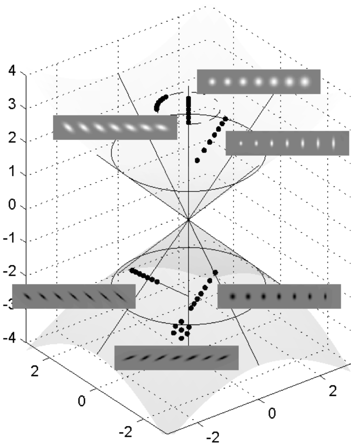


Figure 5: Groups of similar blobs and their representation in the Blobs' Space

4 Blob grouping

As we have seen in previous section groups of blobs with similar properties form meaningful clusters. However, these clusters will not be always given by gaussian distributions in the space, but they can be given by parameterizable shapes, such as rings, lines or planes. This is the reason why we use the Normalized Cuts algorithm [18] as the basic clustering technique to extract the blob groups.

4.1 Normalized cuts

This method is based on taking the collection of points as the nodes of a graph whose edges are the degree of similarity between points, called weights. The method divides the whole set of points into two disjoint sets minimizing the Normalized Cut, which is a measure of the weights that have been removed to perform the cut. A label $y_i \in \{-1, +1\}$ is assigned to each point of the set, corresponding to the two sets that have been formed. The cutting is performed recursively as long as the cost of cutting is less than a fixed threshold.

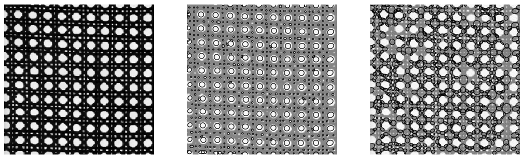
Taking the set of blobs as the nodes of the graph, we need to define the weights w_{ij} between points x_i and x_j. As the space has been designed uniform, the Euclidean distance is used and the weights matrix is defined as $w_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma^2}}$ where x_i is the position of the point corresponding to the attributes of the $i-th$ blob. Assuming the labelling vector y can take real values, it can be obtained by solving a generalized eigensystem. In [19] a version of the Normalized cuts is presented which avoids sensitivity to outliers, which suits better our data.

4.2 Results

As we introduced in section 1 we propose to build a semantic description of a texture image based on the *blob group decomposition*, denoted as BG, that we propose to obtain from the computation of

$$NCut(\mathcal{U}(\mathbf{B}^{\mathcal{I}})) \tag{5}$$

where the resulting clusters can be interpreted as an image decomposition where each group of blobs can have a specific meaning. In figure 6 we show the application of the derived algorithm to the original texture image given in (a) together with its BD in (b) and (c), where bright and dark blobs are separated with a legibility purpose. Its representation in the uniform space and its BG as the result of the NCut algorithm are shown in figure 7.



(a) (b) (c)

Figure 6: BD of original image in (a). Bright blobs are represented in (b) and dark ones in (c)

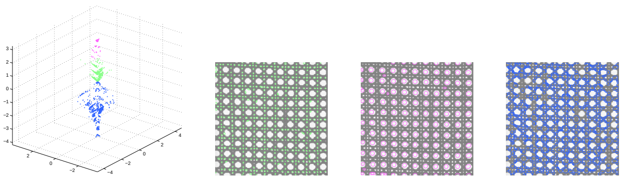


Figure 7: BG of the image in figure 6.(a) on the Blob Space and on the image.

The results show how the method correctly detects there are two kind of bright isotropic blobs and only one kind of dark blobs which are not Gaussian-like but which are well approximated.

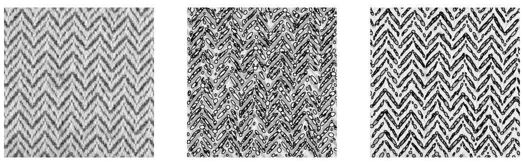
The BG is also computed for another texture image in figure 8. The analysis of the detected blobs in figure 9 demonstrates how the blob detector works in images where blobs are not isotropic and with noise, and how the ellipses (or detected blobs) cover the different kinds of blobs, depending of their attributes. It can also be seen how the orientations of the blobs are well separated.

Despite having filtered the image by isotropic filters, and thus having explored the local grey-level around blob centers in an isotropic neighborhood only, it can be appreciated how big dark blobs are well detected. This is due to the assumption of equal width between the blob and the filter which has detected the presence of a blob.

In fact, elongated blobs with a high aspect ratio are approximated as the addition of elongated blobs with smaller values of ar. The detection of those elongated blobs and therefore the texture decomposition can be improved by introducing a refinement step on the selection of the filters. If the image was filtered by elongated filters with the appropriate orientation the elongated blobs with high values of ar would be better detected.

5 Applications

In this section we want to show three different applications where the proposed algorithm and its results can provide interesting contributions.



(a) (b) (c)

Figure 8: BD of original image in (a). Bright blobs are represented in (b) and dark ones in (c)

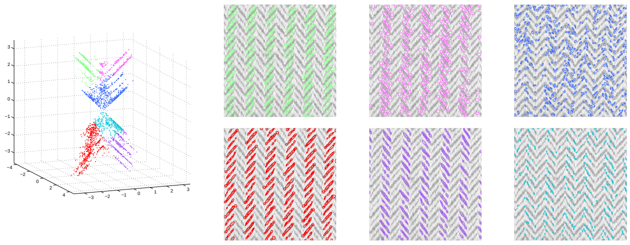


Figure 9: BG of the image in figure 8 on the Blob Space and on the image.

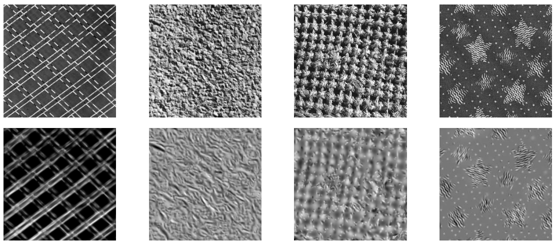


Figure 10: Original Images at the top and the corresponding synthesis from the BD at the bottom.

5.1 Synthesis of texture images

In order to show the completeness of the proposed decomposition we show in several examples the capability of performing image synthesis just from the BD representation. In figure 10 we show four original images on the top, and the corresponding synthesis from the BD derived from each one, at the bottom. The image synthesis is obtained by the addition of Gaussian kernels with the attributes specified by the BD.

The results of the reconstructed images are shown in the same figure, and it can be seen the loss of information of the structure of the image is not important. Furthermore, the irregularities of structured patterns are well maintained in the reconstructed image, which is not common in frequential decompositions of images.

5.2 Coloring gray-level texture images

The information provided by the BG of the image can also be used to color the images according to the attributes of the blob groups. In figure 11 results of this coloration are shown. This application is particularly interesting for image design.

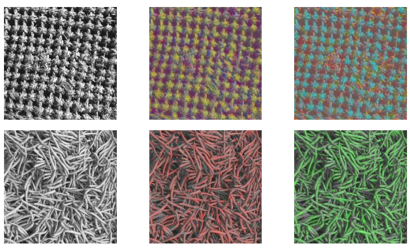


Figure 11: Image colorations based on the BG of the images

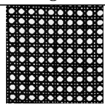
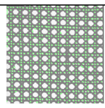
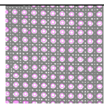
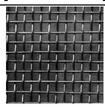
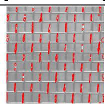
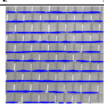
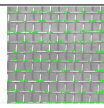
	Original	Blob group decomposition		
				
PBC	[4 1 4 3 3]	[4 1 4 2 2]	[4 1 4 1 1]	
				
PBC	[4 1 4 3 3]	[4 1 · 3 3]	[4 4 · 4 4]	[2 1 · 1 2]

Figure 12: (a)Texture images that share the same PBC descriptor $PBC = [4\ 1\ 4\ 3\ 3]$. (b), (c) and (d) show the images decomposition

5.3 Texture image description based on the BG

As it has been said, the direct correspondence between the BG that is given by the method and a description that could be given by human subjects asked to describe the images in terms of their blobs is one of the most interesting aspects of the texture image representation presented here.

Given the BG of an image, its description can be given in terms of the features of the blobs forming each group, so that the size of the description for an image depends on the number of different characteristic elements forming the texture or its complexity.

Among the descriptors presented for the MPEG-7, the more similar to our goal is the Perceptual Browsing Component [5]. This descriptor has 5 components: one for regularity, two for the dominant orientations and two for the dominant scales. Even if this descriptor is known to be useful for texture browsing it still has some limitations due to the fact that only 5 components are taken into account, with a few possible values each. For instance, the two textures shown in figure 12 have the same description, even though one is basically formed out of circular blobs and the other has none. Hence, the PBC gives the same description for textures that would receive different descriptions by users.

In figure 12 the BG of bright blobs of two images sharing the same PBC are shown. The values of the PBC for the original images and for each image corresponding to a blob group are given below. The description obtained from the BG is more complete than the one given by the PBC solely. For instance, the decomposition of the first image shows that both sizes of bright blobs have the same orientation, due to the position of the blobs. On the other hand, for the second image it is shown the two main orientations correspond to different types of blobs. It has to be taken into account only part of the information contained in the BG is considered, and a new descriptor including the shape of the blobs or more information about their contrast could be built.

6 Discussion

A uniform blob space where distances are meaningful from a perceptual point of view has been presented. This space is useful as a basis to group similar blobs in texture images, which can provide an explanation of the image in terms of its attributes. Experimental results on texture image synthesis show the completeness of the proposed decomposition. Therefore, we can conclude this image representation presents two interesting properties: it is compact in terms of space cost, and it is non-redundant since most of the image points belong to a low number of image blobs whose attributes are known. This is an advantage with respect to other wavelet or filter-based decompositions where redundancy is a common problem.

As further research we propose this uniform space and the grouping process as a previous step for a filter selection procedure that can provide the attributes to design a set of filters which can improve the detection of elongated blobs, which is still a drawback in general blob detection. The improvement on the blob detection can allow to extend applicability to other visual tasks, such as saliency detection.

7 Acknowledgements

This work was partially supported by the project TIN2007-64577 funded by Ministerio de Educación y Ciencia.

References

[1] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R.Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. on PAMI*, 22(12), 2000.

[2] Manjunath, Salembier, and Sikora, editors. *Introduction to MPEG-7.Multimedia Content Description Interface*. John Willey and Sons, 2002.

[3] A.R. Rao and G. L. Lohse. Towards a texture naming system: Identifying relevant dimensions of texture. *Vision Research*, 36:1649–1669, 1996.

[4] F. Liu and W. Picard. Periodicity, directionality, and randomness: Wold features for image modelling and retrieval. *IEEE Trans. on PAMI*, 18(7):722–733, 1996.

[5] P. Wu, B. S. Manjunath, S. Newsam, and H.D. Shin. A texture descriptor for browsing and similarity retrieval. *Journal of Signal Processing: Image Communication*, 16:33–43, 2000.

[6] B. Julesz and J.R. Bergen. Textons, the fundamental elements in preattentive vision and perception of textures. *Bell Systems Technological Journal*, 62:1619–1645, 1983.

[7] Y. Wang S.-C. Zhu, C. Guo and Z. Xu. What are textons ? *International Journal on Computer Vision*, 62, 2005.

[8] T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, 1994.

[9] D. Marr. *Vision*. W. H. Freeman and Company, 1982.

[10] H. Greenspan C. Carson, S. Belongie and J. Malik. Blobworld : Image segmentation using expectation-maximization and its application to image querying. *IEEE Trans. on PAMI*, 24(8), 2002.

[11] T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3-d depth cues from affine distortions of local 2-d brightness structure. *Image and Vision Computing*, 15, 1997.

[12] A. Almansa and T. Lindeberg. Fingerprint enhancement by shape adaptation of scale-space operators with automatic scale selection. *IEEE Transactions on Image Processing*, 9(12), 2000.

[13] L. Bretzner and T. Lindeberg. Qualitative multi-scale feature hierarchies for object tracking. *Journal of Visual Communication and Image Representation*, 11, 2000.

[14] G. Wyszecki and W.S. Stiles. *Color science: concepts and methods, quantitative data and formulae*. John Wiley & Sons, 2nd edition, 1982.

[15] M. J. Morgan. The visual computation of 2-d area by human observers. *Vision Research*, 45, 2005.

[16] J.M. Zanker and T. Quenzer. How to tell circles from ellipses: perceiving the regularity of simple shapes. *Naturwissenschaften*, 86, 1999.

[17] D. Regan and S.J. Hamstra. Shape discrimination and the judgement of perfecte symmetry: dissociation of shape from size. *Vision Research*, 32, 1992.

[18] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on PAMI*, 22(8), 2000.

[19] A. Rahimi and B. Recht. Clustering with normalized cuts is clustering with a hyperplane. In *Workshop on Statistical Learning in Computer Vision*, 2004.