A Retina-like Image Representation of Primal Sketch Features Extracted using a Neural Network Approach

Herman M Gomes, Robert B. Fisher and John Hallam
Department of Artificial Intelligence, The University of Edinburgh
5 Forrest Hill, Edinburgh EH1 2QL, Scotland

Abstract

This paper presents a log-polar image representation composed of low-level features extracted using a connectionist approach. The low level features (*edges*, *bars*, *blobs* and *ends*) are based on Marr's primal sketch hypothesis for the human visual system [3] and are used as the entry point of an iconic vision system [1]. This unusual image representation has been created using a neural network that learns examples of the features in a window of receptive fields of the image representation.

1 Introduction

Traditional image feature extraction operators have usually been designed by hand, work independently of each other and act on Cartesian images (an artifact of sensor architecture). However, the architecture of the primate vision system seems to be quite different, and we can use this to produce interesting results in artificial vision systems.

The outermost retinal region is formed by rings with a fixed number of receptive fields whose distance from the retina centre can be expressed in terms of an exponential function [6]. The mapping from this region to the visual cortex can be mathematically approximated by a log-polar representation, which transforms both rotation and scaling in the Cartesian domain into translation in the log-polar domain and cuts off most of the complexity involved when recognising objects at different scales and orientations.

The iconic vision system described in [1] is a typical system which uses the log-polar approach. The input to the system is a set of primal sketch planes which, in a first version of the system, were extracted using heuristically defined logical operators. Another example is given in [4], where a vision architecture is defined in terms of a log-polar representation with a Difference of Gaussians as the receptive field function and a template matching algorithm operating directly on the log-polar space.

This paper presents a log-polar image representation composed of low-level features extracted using a different approach. The low level features (*edges*, *bars*, *blobs* and *ends*) are based on Marr's primal sketch hypothesis for the human visual system [3]. The primal sketch represents a more compact representation for the image data and provides cues for an attention mechanism under the experimental evidence that these kinds of low level features seems to attract visual attention [7].

Instead of trying to build a model for completely describing the features, which is a complex task because of the unusual geometry of the image data and the receptive field integration, learning the features was a sensible option. In our work, a neural network approach was chosen due to its adequacy when learning data in which there is no obvious structure or logical organisation.

2 Image Representation

The input image is resampled through the use of a mask consisting of concentric rings of overlapping circular receptive fields whose centres are geometrically spaced from the centre of the mask. If we define an image which is accessed by using the rings (logarithm of the distance of the rings to the retina centre) and sectors of the previous mask, then we have a log-polar image representation. The innermost mask region, named the fovea, contains receptive fields hexagonally distributed with a high density. We have simulated a hexagonal packing outside the fovea by shifting each consecutive ring by half of the angle defining a sector of receptive fields. The radius of the n^{th} outer retinal layer is: $R(n) = \beta^n R(0)$, where $R(0)$ is the radius of the first layer of receptive fields exterior to the fovea and β defines the geometrical progression of distances of receptive field layers from the retinal centre (we have used $\beta \approx 1.1$). Similarly, the radius $r(n)$ of a particular receptive field in layer n is $r(n) = \beta^n r(0)$.

We have defined the fovea as having 11 layers of receptive fields. Each receptive field in the fovea has a radius of 0.5 of a pixel. Outside the fovea, there are 33 more layers of receptive fields distributed accordingly to the previous equation. Each receptive field overlaps with each of its neighbours by approximately 53% of its diameter. These parameters produced a retinal mask with a diameter of 256 pixels which covers a circular region of the Cartesian image.

2.1 Estimating the reflectance information

The output of a given receptive field is calculated according to the equation:

$$O = \sum_{x^2+y^2 \leq r^2} I(x,y)F(x,y) \tag{1}$$

where O is the neuron output, $I(x,y)$ is the perceived intensity and $F(x,y)$ is the receptive field function, both applied to points (x,y) in the receptive field circular domain of radius r. One of the simplest function that can be used to simulate the receptive field computation is the uniform averaging function [1]. However, most of the biological processes are better expressed in terms of non-linear functions. A non-linear alternative investigated in this work is the normalised Gaussian function.

The human visual system computes an approximate lightness constancy, and this implies that it is somehow able to extract information about the invariant surface reflectance of objects (almost) independently of changes in illumination and scene composition [2]. We have used a model for estimating the original reflectance information from the objects (see Eq. (2)), which is derived from the receptive field computation. The $log(E)$ term in Eq. (2) is nearly constant over local image regions and then makes the receptive field computation

O' a good approximation for the weighted logarithm of the reflectance.

$$O' = log(E) + \sum_{x^2+y^2 \leq r^2} log(R(x,y))F(x,y) \qquad (2)$$

where E is the irradiance falling on the object, R is the local surface reflectance.

3 Feature Extraction

Features are trained and detected in a window of receptive fields. We have chosen a window composed of a central receptive field plus its next 6 and 12 surrounding neighbours, totalling a sum of 19 receptive fields hexagonally distributed. For training and testing purposes, synthetic exemplars of the features are drawn in a fixed position of the input image corresponding roughly to a particular 19-window of receptive fields. Then, the output of all of these 19 receptive fields is computed and used as input for the neural network modules. As the receptive fields have variable sizes throughout the retinal image, one could think that different trainings would be needed for each scale of the receptive fields. However, the receptive field computation produces normalised values once it applies a function whose integral is always 1 independently of the area.

3.1 Training

Several different combinations of contrast, orientation, size, and noise level were used. Table 1 summarises the parameters used in the generation of the training sets.

Feature	Orientation	Contrast	"Size"	Noise (σ)
Edges	0, 30, ..., 330°	0.5, 0.75, 1.0	$\otimes$	0, 5
± *Bars*	0,30, ..., 150°	0.5, 0.75, 1.0	0.8, 0.9, 1.0	0, 5
± *Blobs*	$\otimes$	0.5, 0.75, 1.0	0.8, 0.9, 1.0	0, 5
± *Ends*	0,30,..., 330°	0.5, 0.75, 1.0	30°, 45°	0, 5

Table 1: Parameters used in designing the training sets. $\otimes$=not applicable.

Network architecture and training parameters. We have chosen the backpropagation algorithm minimising a least square error metric, because of its simplicity and reasonable computational power. We have used an approach which minimises the problem of having large training sets and uses a disposition of output neurons which is favourable to coarse coding.

Three of the feature classes (*bars*, *blobs* and *ends*) were partitioned into 6 new feature classes according to their intensity. Whenever a particular feature was drawn with a lighter intensity, compared to the background, it was named *positive*, *negative* was the reverse. Then, seven different neural modules, each one designed for a particular feature, was built. To provide counter examples for each class, random features and features from the other classes were used. Table 2 contains some examples of the training features.

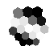
	Edge	*+ Bar*	*- Blob*	*+ End*	*Counter* Example
Cartesian Inputs					Not applicable
Retinal Outputs					

Table 2: Some examples of the training features.

All of the modules have an input layer composed of 19 neurons, followed by a hidden layer. Finally, there is an output layer formed by neurons associated to each of the 6 or 12 standard orientations and a last neuron associated to an unknown class. The strength of response of an output neuron is a function of the feature's contrast as explained below. In the case of the positive and negative *blobs* the network's output layer has only 2 neurons, one coding the *blob* itself and the other coding the non-blob class.

We have used a learning rate of 0.01, inertia or momentum of 0.95 and a neural module was considered trained when all the training patterns passed with 0.05 error bound. Table 2 summarises the class specific architectures and training parameters.

Neural Module	Architecture $I \times H \times O$	# of Epochs $\ominus, \oplus$
Edge	$19 \times 16 \times 13$	12040
Bar	$19 \times 13 \times 7$	24030, 23610
Blob	$19 \times 8 \times 2$	12210, 21550
End	$19 \times 16 \times 13$	28380, 34330

Table 3: Training parameters. Legend: $I \times H \times O$=Input, Hidden and Output units. $\ominus, \oplus$ = Number of training epochs corresponding to the sets having positive and negative features, respectively, not applicable for *edges*.

Coding the contrast information. The contrast within a retinal window is calculated according to the Eq. (3). The desired output for a neuron representing a particular feature was represented in terms of a function of the contrast. We have used the *Weibull* function, Eq. (4), which is an approximation for the psychometric contrast detection function in humans [5].

$$c = \frac{|L_{max} - L_{min}|}{L_{max} + L_{min}} \tag{3}$$

$$W(c) = 1 - exp\left[-\left(\frac{c}{\alpha}\right)^{\beta}\right] \tag{4}$$

where L_{max} and L_{min} are the minimum and maximum intensities found in an image patch, respectively. In order to have the output of the Weibull function in the range $(0, 1)$ we have chosen $\alpha = 0.5$ and $\beta = 3$.

3.2 Initial results

The results of the application of the trained neural modules over a test image are shown in Appendix A. The retinal mask was applied to the Cartesian image, then all the possible 19-receptive field windows were extracted and fed into each of the trained neural network modules. The "winner-takes-all" strategy was used when reading the classification results from a module's neuron outputs. Finally, all the classified central receptive fields of which the winning neurons produced an output smaller than a rejection threshold where discarded.

Intuitively, it is possible to say that the detection of *edges* presented the best results. *Bars* can be ranked in a second place. And finally, *ends* and *blobs* presented the worst results. A possible cause to these latter results can be associated with the way the synthetic features were designed: a positive blob, for instance, has been drawn in the Cartesian domain as a bright central region surrounded by a darker uniform region, but real blobs might be surrounded by a region of pixels of non-uniform intensities. It is possible to sort out the above problem by adding features manually extracted from real images to the training sets of synthetic features (currently under implementation).

4 Conclusions and future work

According to the results we can conclude that it is feasible to learn primal sketch features in a retina-like image representation. However, it is not yet clear how could we access the practical results of the trained networks over real images. Features are detected at different scales, and so, a blob seen in the peripheral region of the retina can be classified as a ring of edges if it were instead seen nearby the foveal area, because of the varying size of the receptive fields, but this is also true in humans. It is important to note that these feature detectors operate in a different image structure and therefore they will not detect the same features that conventional feature detectors would find in Cartesian images. Our next target will be to assess the effectiveness of the presented technique when compared to the previous approach described in [1]. This will also include the investigation of parameters that might play a role in the recognition process, like those defining the retina structure, the receptive field function and the feature's window size.

References

[1] T. D. Grove and R. B. Fisher. Attention in iconic object matching. In R. B. Fisher and E. Trucco, editors, *7th British Machine Vision Conf.*, volume 1, pages 293–302, 1996.

[2] A. C. Hurlbert. Neural network approaches to color vision. In H. Wechsler, editor, *Neural Networks for Perception*, volume 1(Human and Machine Perception), chapter II.5, pages 265–284. Academic Press, 1992.

[3] D. Marr. *Vision.* W. H. Freeman and Co., 1982.

[4] J. P. Siebert and I. Eising. Scale-space recognition based on the retino-cortical transform. In *Proc. IEE Conference on Image Processing and its Applications*, 1995. Edinburgh.

[5] R. Watt. *Understanding Vision*. Academic Press, 1991.

[6] S. W. Wilson. On the retino-cortical mapping. *Int. J. Man-Machine Studies*, 18:361–389, 1983.

[7] A. L. Yarbus. *Eye Movements and Vision*. Plenum Press, 1967.

A Extracted features

Features of a given test image extracted using the neural network approach are presented below. The black spots are the centres of receptive field windows representing the features. The following rejection thresholds were used in this example: *edges*: 0.05, *bars*: 0.1, *blobs*: 0.3 and *ends*: 0.3.