A RETINAL IMAGE PROCESSING SYSTEM

by

NADER RIAHI

**A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE**

in

THE FACULTY OF GRADUATE STUDIES

Electrical Engineering

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

8 October 1987

© Nader Riahi, 1987.

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical Engineering

The University of British Columbia
1956 Main Mall
Vancouver, Canada
V6T 1Y3

Date OCT. 15 / 1987

ABSTRACT

It was desired to design an image processing system that resembled the human retina, for the processing of two-dimensional images. In particular the system was required to carry out basic image processing tasks such as edge detection.

A new filtering technique was deduced from the physiological findings on the distribution of the receptive fields of the ganglion cells on the retina. This filtering technique was then incorporated in designing an image processing system in which the spatial resolution was increased linearly towards the geometrical center of the image.

The design was based on a discrete distribution of processing areas on an inhomogeneous hexagonal sampling grid. This resulted in a highly localized processing system which simplified the development of algorithms for higher image processing tasks such as boundary following.

The retinal image processing system was simulated on the VAX-11/750. The computational cost of conducting operations such as edge detection, boundary detection and boundary following, using the designed system, was evaluated and compared with that of the conventional image processing system.

TABLE OF CONTENTS

ABSTRACT	ii
LIST OF TABLES	iv
LIST OF FIGURES	v
ACKNOWLEDGEMENT	vii
1. INTRODUCTION	1
1.1. MACHINE VISION SYSTEMS	1
1.1.1. IMAGE ACQUISITION	1
1.1.2. IMAGE PROCESSING	3
1.2. HUMAN VISUAL SYSTEM	5
1.2.1. IMAGE ACQUISITION	5
1.2.2. IMAGE PROCESSING	8
1.3. A MODEL OF THE HUMAN RETINA	10
1.3.1. FILTERING OF IMAGES	12
1.3.2. A NEW FILTERING TECHNIQUE	13
1.3.3. A MODEL OF THE HUMAN RETINA	15
1.3.4. SUMMARY OF THE PROPERTIES OF THE PROPOSED IMAGE PROCESSING SYSTEM	16
1.4. REVIEW OF THE RELEVANT LITERATURE	17
2. MATHEMATICAL MODELING OF A RETINAL PROCESSING SYSTEM ...	18
2.1. DESIGN CONSIDERATIONS	18
2.2. DESIGN OF THE PROCESSING GRID	20
2.2.1. PERIPHERAL REGION	20
2.2.2. FOVEAL REGION	29
2.2.3. SEMI-PERIPHERAL REGION	30
2.3. PROCESSING OF IMAGES	31
2.3.1. IMPULSE RESPONSE OF THE PROCESSORS	31
2.3.2. PROCESSING ALGORITHM	34
3. EVALUATION OF A RETINAL PROCESSING SYSTEM	38
3.1. DESCRIPTION OF THE PROCESSING GRID	38
3.2. THE RESULTS OF BAND-PASS FILTERING	41
3.3. EDGE DETECTION IN THE IMAGE	43
3.4. EDGE FOLLOWING IN THE FILTERED IMAGE	48
3.5. BOUNDARY FOLLOWING IN THE PROCESSED IMAGE	54
3.6. COMPUTATIONAL COST OF RETINAL PROCESSING	57
3.7. ERROR ANALYSIS	61
4. SUMMARY AND CONCLUDING REMARKS	69
4.1. EXTENSIONS TO THIS PROJECT	70
REFERENCES	73

List of Figures

Figure 1, The Laplacian of a Gaussian and its response to step change in intensity	5
Figure 2, The spatial sensitivity profile of a Difference of two Gaussian function	10
Figure 3, The spatial arrangement of processing areas for complete coverage of an image with minimum number of processing areas	20
Figure 4, The position of the processing areas with respect to the associated eccentricities	22
Figure 5, The position of the processing areas with respect to the associated eccentricities, when the arrangement of Fig. 4 is rotated by 60 degrees around the center of the processor area at E	24
Figure 6, The spacing of the neighbouring processing areas with respect to their radii and the associated eccentricities	26
Figure 7, The spatial sensitivity profile of a DOG function and the associated stripes for the local coordinate transfer	35
Figure 8, The distribution of the centers of the processing areas	41
Figure 9, The raw image on a rectangular sampling grid	43
Figure 10, The result of the filtering of the image in Fig. 9, based on the grid shown in Fig. 8	43
Figure 11, The detected edges in the filtered image shown in Fig. 10	48
Figure 12, The binary equivalence of the image shown in Fig. 11, where all edges of amplitude greater than 2 were set to amplitudes of 255 ...	48
Figure 13, Result of applying dynamic programming to the image shown in Fig. 12, for grouping of edges to form boundary segments	54
Figure 14, Result of boundary following algorithms for the detection of the boundary of the hat of the girl shown in Fig. 9	57
Figure 15, Direction of the movements of the centers of the processing areas 1 and 2 due to the quantization of the coordinates	63
Figure 16, Position of the centers of the processing areas after quantization. The lines between the dark and light regions determine the position of the edges	65

Figure 17, The sign representation of the values of the sample points after the filtering process. Each sample point corresponds to the center of the associated processing area. the arrows show the orientation of the detected edges at point 1 65

List of Tables

The local direction of the target pixels with respect to the center pixel $[i,n]$... 46

ACKNOWLEDGEMENT

I would like to use this opportunity to thank Dr. P.D. Lawrence for his valuable technical advice and support throughout the development of this work. I would also like to thank my brothers for their moral support and concern, in particular Mr. I. Riahi for his continual financial support throughout the course of my Masters degree. Last but not least, I would like to thank Miss J. Krouger for her sincere help in typing of this report.

I dedicate this work to my parents.

1. INTRODUCTION

1.1. MACHINE VISION SYSTEMS

Conventional machine vision systems are generally partitioned into three distinct sections:

1. image acquisition,
2. image processing, and
3. image analysis.

Recognition of objects is an example of image analysis. The recognition task however requires extensive computation which is not only time consuming but requires sophisticated and expensive hardware. It is the purpose of this report to introduce a new method of image preprocessing to reduce the computational cost of employing higher decision making algorithms. The new method is based on the studies of the human vision system and will be explained in Section 1.3.

1.1.1. IMAGE ACQUISITION

Starting from the natural scene, the light rays are collected by the lens and brought together in the image plane. †The image is then sampled by an array of photodetectors. Photodiodes are one example of many different types of photodetectors. Basic properties of a photodiode are:

1. the light absorbing surface is flat.
2. the light sensitivity is generally constant over the absorption surface
3. the output is a time varying electronic signal whose amplitude is proportional to the average of the light intensity over the absorption

† This report concentrates only on two dimensional images of three dimensional scenes.

surface.

The geometrical distribution of these photodiodes on the image plane represents the sampling grid with which the image is sampled. Many different types of sampling grids are possible but this report is mainly concerned with a rectangular array of sensors.

Sampling

According to the Nyquist sampling theorem[R. M. Mersereau], as long as the image contains no spatial frequencies greater than half the sampling frequency then the underlying image can be unambiguously represented by its samples. It is however noteworthy that natural scenes are not band-limited and that their spatial frequency spectrum stretches to infinity. If the effect of the lens and aperture is ignored, then since the scene is not band-limited, its image will not be band-limited either. This means that the sampling of the image will result in some aliasing effects. The sampling theorem however is based on infinitesimally small sample points which is not applicable in cameras. the reason being that:

1. photodetectors have a finite surface area, and
2. output of a photodetector is proportional to the average of the light intensity over its surface area.

Averaging is one form of low pass filtering. As a consequence, the high frequency contents of the image are being low pass filtered and therefore aliasing is reduced.

There are many strategies by which sampling of a two dimensional image can be performed [E. Dubois]. The most common approach is rectangular sampling. With rectangular sampling, a band limited image is sampled at evenly spaced

values of two independent (orthogonal) variables. It has been the method of choice for a variety of reasons:

1. algorithms for processing signals which have been rectangularly sampled can be straightforwardly generalized from the one-dimensional case,
2. the resulting expressions can be readily understood and implemented in software,
3. the hardware to perform the sampling is straightforward to build.

Unfortunately the rectangular sampling is not the most efficient strategy as far as the resulting signal processing algorithms are concerned [R. M. Mersereau].

1.1.2. IMAGE PROCESSING

After the image is sampled by an array of sensors, signals from each sensor are read out by some scanning mechanism and quantized according to the number of brightness levels required. The image is therefore stored as an array of numbers representing the intensity of the image at the corresponding points. The raw image data however is not very useful for higher processing (such as object specification), due to its dependence on the lighting condition and view point. Furthermore there is much spatial redundancy in the raw data since the neighbouring samples have the same or nearly the same values. A collection of techniques exploit this redundancy in order to undo the above degeneracies [Duda and Hart , Young and Fu, Ballard and Brown]. These techniques have the character of transforming the raw image data into parameters such as local edges.

Edges

Edges in an image are defined as an abrupt change in the intensity. Being more

interested in intensity changes rather than intensity values, vision systems must have some means of obtaining the location of edges along with additional information such as the local direction of the edge and its strength along that direction. Also a knowledge of the scale at which these measurements are made can be of great use [Marr, pp. 46]. In general, a filter that suppresses uniform intensity profiles and enhances the intensity changes in an image is required. Differential operators seem to incorporate these specifications. The derivative of a step change in the intensity will have a peak value proportional to the strength of the edge, occurring at the location of the edge. For an ideal step change however, the differential operator will result in a spike which is undesirable. This is why some degree of smoothing is required to suppress the high frequency contents of an edge. Also it is troublesome to locate these peaks in the filtered image, and for this reason a second order differential operator is preferable. The second derivative of a step change in intensity will have a level crossing occurring at the location of the edge with a maximum and a minimum at equal distances on opposite sides of it. The value of these peaks and ridges is proportional to the strength of the edge.

Filtering

Marr and Hildreth suggested that the Laplacian of a Gaussian was the best fit to these specifications. Lunscher [Lunscher and Beddoes (I) and (II), 1985] has also conducted some work on the optimality of Laplacian of a Gaussian for edge detection in the natural environment. The Gaussian part of the filter plays an important role in the processing of images. The standard deviation of this Gaussian determines the degree of smoothing and hence provides some indication of the spatial frequencies resolvable by the filter. The spatial profile of this filter

along with its response to a step change in the intensity is shown in Figure 1. Filtering takes place by convolving this operator with the image. The output of the convolution carries all the information about the intensity changes at a particular scale and can be used for higher processing stages such as stereopsis, boundary detection, object recognition etc.

1.2. HUMAN VISUAL SYSTEM

Biological vision systems may be interpreted as some form of hierarchical conversion system, where optical patterns are converted into electrical signals and processed as they are transferred through complex neural connections. It is the purpose of this section to describe what is known about the individual stages of this hierarchical system and explain the particular type of processing that is performed at each stage. As in the case of machine vision systems, human vision system can be partitioned into:

1. image acquisition, and
2. image processing

The image processing however is followed by an elaborate system of neural connections that perform decision-making. It is these neural connections that constitute the brain and its behavior [S. W. Kuffler].

1.2.1. IMAGE ACQUISITION

Light patterns enter the cornea and pass in sequence through the anterior chamber, the pupil opening of the iris, the lens and the vitreous humour, before impinging on the layer of photoreceptors which constitute the retina at the rear of the eye. It is the retinal receptors that are responsible for the conversion of

(a) The spatial profile of a Laplacian of a Gaussian

(b) The spatial profile of a step change in intensity

Figure 1, The Laplacian of a Gaussian function and its response to a step change in intensity

light patterns to electronic signals in the form of frequency modulated pulses [M.D. Levine].

Photoreceptors

In the human eye there are two types of photoreceptors, rod cells and cone cells. Rods can mediate vision in dim light but are so sensitive that they become saturated in normal daylight. Daylight vision is mediated by cones which can operate successfully in high light levels. This report only concentrates on the cone cells. At one end of the cell is the outer segment surface which absorbs the light and on the other end of the cell is the synaptic ending which relays the generated electrical signals to other neurons, namely bipolar and horizontal cells. The absorbing surface of the cone is of conical shape, hence the name cone. It is important to know that the size of the cones vary and become larger towards the peripheral regions of the retina [S.W. Kuffler, D.R. Williamss]. The importance of the shape and size variance of the cones become clear when considering the sampling grid by which the images are sampled in the eye.

Sampling

It is well established that cone cells on the retina are distributed in a hexagonal array [R.E.Steinberg, M. Reid, P.L. Lacy, D.R. Williamss] and that the cone spacing (sampling frequency) is not constant over the whole retina. The cone spacing increases towards the peripheral region of the retina. This kind of sampling is generally referred to as *inhomogeneous hexagonal sampling*. As mentioned in Section 1.1.1, aliasing is a direct result of under sampling and to avoid the problems caused by aliasing, the image must be band limited to frequencies lower than half the sampling frequency. As a consequence of inhomogeneity of the sampling grid on the retina, one would expect to have an

increasing aliasing effect towards the peripheral region. This however is not the case in the human eye, since the size of the photoreceptors become larger in the peripheral region of the retina. Therefore the images are being low pass filtered to a greater extent and hence the aliasing effects stay the same. The shape of the cones play an important role when considering the peripheral region of the retina. This is explained as follows: if the absorption surface of the cone cells were flat then the low-pass filtering behavior of the large cones in the peripheral region of the retina would be less ideal (more Gibbs phenomenon is produced [Oppenheim]). Integration of light intensity over the cone's conical surface however, is equivalent to integrating the light intensity with a conical light sensitivity profile across a flat surface. This kind of sensitivity profile is a better low pass filter than a simple averaging over the same surface area [D.R. Williams].

1.2.2. IMAGE PROCESSING

The human eye not only captures images but also performs preliminary processing on the images. The preliminary processing includes noise reduction and feature extraction. This means that only the useful information in the image is passed through the optic nerves to the higher processing cells and the redundant information is filtered out.

Processing cells

As it was explained before in Section 1.2.1, the retina is not just a mosaic of photoreceptors. There are interactions between neighbouring cones, some of which is achieved by the horizontal cells. The output of a number of cones are transmitted through synaptic gaps to the bipolar cells. The bipolar cells

themselves interact with each other by means of amacrine cells. The output of a number of bipolar cells in turn is transmitted to ganglion cells. Ganglion cells are thought to perform the first stage of processing on the image. They behave like filters, suppressing noise and uniform intensity profiles and responding if there is a change in the intensity [M.D. Levine]. Their function can be related to edge extraction in machine vision system. The output of a ganglion cell is a frequency modulated signal. The strength of the output signal is a measure of the amount of the intensity change in the image. The ganglion cell density decreases towards the peripheral region of the retina, just as the cone density does [J. Stone]. Ganglion cells are highly localized and only fire if there is a change in the intensity of the image within their operating area. This operating area is referred to as the receptive field of a ganglion cell.

Receptive field

It is evident from the foregoing that a number of photoreceptors participate in the stimulation of a single ganglion cell. The area covered by these photoreceptors is called the receptive field of a ganglion cell. The receptive field of the majority of ganglion cells are concentrically organized into centers with antagonistic surrounds. The photoreceptors in the central section of the receptive field have an excitatory effect on the firing of the ganglion cell and the receptors in the surround section of receptive field have an inhibitory effect. This type of organization is referred to as an ON-center ganglion cell. The size of the receptive fields of the ganglion cells increase towards the peripheral region of the retina [L. Peichl, H. Wässle]. The number of photoreceptors in the receptive field of a ganglion cell is referred to as the receptor-ganglion cell convergence. This convergence value increases towards the central region of the retina (fovea).

Filtering

The spatial sensitivity profile of the receptive fields of ganglion cells are approximately the shape of a DOG (Difference Of two Gaussians) function [D. R. Williams]. This is shown in Figure 2. The diameter of the central region of this function is related to the space constants of the Gaussians. The DOG function is a close approximation to the Laplacian of a Gaussian [Marr and Hildreth]. This means that after convolving a step intensity profile with the DOG function, a zero-crossing is generated where the position of the zero-crossing corresponds to the position of the intensity change. Therefore the ganglion cells can be considered as edge detectors with their impulse responses defined by the spatial sensitivity profiles of the receptive fields.

Overlap factor

It was mentioned before that ganglion cells are highly localized. This means that the ganglion cells at different points on the retina operate with no interactions. This is only true in a very global sense. Locally there is an overlap between the receptive fields of neighbouring ganglion cells. The number of ganglion cells that are stimulated when the size of the stimulus is equal to the size of the receptive field center, is referred to as the overlap factor. The overlap factor is independent of the position on the retina [B.Fischer].

1.3. A MODEL OF THE HUMAN RETINA

The physiology of the eye and the type of processing that is carried out by the eye were explained in Section 1.2. It is the purpose of this section to explore the advantages of the human vision system over the conventional machine vision system as far as the image processing is concerned. A new filtering technique

Figure 2, The spatial sensitivity profile of a Difference of two Gaussian function

based on the physiology of the eye is proposed. The new filtering technique is then utilized for constructing a model of the human retina.

1.3.1. FILTERING OF IMAGES

Images are filtered to extract useful information such as local edges (intensity changes). These edges can then be utilized for a variety of tasks such as boundary detection, texture analysis etc. The filtering process is mathematically represented by the convolution of a filter kernel with the image.

Convolution

The process of convolution is generally partitioned into three steps,

1. multiplication,
2. addition,
3. shift.

In digital filtering for example, each sample value in the filter kernel is *multiplied* by the corresponding sample values in the image. The result of the multiplication for all the corresponding sample points are then *added* together. The filter kernel is then *shifted* by one sample point and the above steps are repeated. This means that the filtered image will have the same number of sample points as the raw image. In the human eye however, the number of sample points in the raw image and the filtered image are not equal. This is explained as follows: It was shown in Section 1.2.2 that the ganglion cells behaved like filters. The impulse response of a ganglion cell is given by the spatial sensitivity profile of its receptive field. Referring back to the convolution process, the cones represent the image sample points, receptive fields represent the filtering kernel and the ganglion cells represent the filter output points.

Physiological findings on the density of the ganglion cells in the retina show that the cone-ganglion cell convergence decreases towards the peripheral region of the retina (Section 1.2.2). This means that there are less sample points in the filtered image than there are in the raw image. Hence the filtering process in the eye not only extracts some information about the intensity changes in the image but also locally reduces the number of sample points at the same time. This combined process of filtering and resampling at a lower rate can be utilized as a new filtering technique in machine vision systems.

1.3.2. A NEW FILTERING TECHNIQUE

For the purpose of edge detection, both the processes of low-pass filtering and sampling rate reduction can be combined into one single process as explained below. Consider a continuous signal $f(x)$ and a low pass filter $h(x)$ with its cut-off frequency ω_c . The output of the filter $g(x)$ is given by the convolution of $f(x)$ and $h(x)$:

$$g(x) = \int f(\lambda).h(\lambda - x) d\lambda \quad (1.1)$$

According to sampling theory, $g(x)$ can be sampled with a sampling frequency of at least $2\omega_c$. If the sampling period (distance between sample points) is given by T , then the sampled output signal is given by:

$$g(n) = g(x).\delta(x - n.T) \quad (1.2)$$

where

$$n = 0, 1, 2, \dots, N$$

N = total number of sample points in the image

Equations (1.1) and (1.2) can be combined to give:

$$g(n) = \int f(\lambda). h(\lambda-x).\delta(x-n.T) d\lambda \quad (1.3)$$

The expression in the right hand side of the equation (1.3) is equal to zero for all values of:

$$x \neq n.T$$

It is obvious that there is no gain in calculating the values of $g(x)$ for those points that will eventually be ignored by the sampling process. Hence Equation (1.3) can be simplified to:

$$g(n) = \int f(\lambda).h(\lambda-n.T) d\lambda \quad (1.4)$$

The same argument can be applied to digital filters. If the input signal is represented by $f(n)$ and its sampling frequency by ω_{s1} , the low pass filter by $h(n)$ and its cut-off frequency by ω_c , then the output of the filter can be resampled at a lower rate as long as ω_c is less than half ω_{s1} [L.O'Gorman and A.C. Sanderson]. The process of low pass filtering and resampling of digitized images can be combined into a single step as in the case of continuous images. The output signal is given by:

$$g(n) = \sum f(k).h(k-n) \quad (1.5)$$

where

$$n = 0, 1, 2, \dots, N$$

$$k = 0, 1, 2, \dots, K$$

N = total number of sample points in the raw image

K = total number of sample points in the filter kernel

It should be noted that the total number of multiplications in Equation (1.5) is given by:

$$M = N.K \quad (1.6)$$

If the cut-off frequency of the filter is ω_c and the sampling frequency ω_{s1} , as long as ω_c is less than $\omega_{s1}/2$, then $g(n)$ can be resampled at a lower rate than ω_{s1} . The new sampling frequency is given by:

$$\omega_{s2} = 2 \omega_c$$

If ω_{s1}/ω_{s2} is an integer then the resampled signal is given by:

$$g(m) = \sum f(k).h(k-(\omega_{s1}/\omega_{s2}).m) \quad (1.7)$$

where

$$m = 0, 1, 2, \dots, (\omega_{s2} / \omega_{s1}).N$$

$$k = 0, 1, 2, \dots, K$$

It should be noted that the total number of multiplications in this case is given by:

$$M = (\omega_{s2} / \omega_{s1}).N.K \quad (1.8)$$

For a typical value of $(\omega_{s2} / \omega_{s1}) = 0.5$ there will be 50% reduction in the number of multiplications.

1.3.3. A MODEL OF THE HUMAN RETINA

It is now possible to construct a simple model of the human retina using the physiological information given in Section 1.2 and the filtering technique explained in Section 1.3.2. The most important feature of the model is that it is

foveocentric. This means that the resolution with which the image is filtered increases towards the geometrical center of the image (fovea). The resolution with which the image is filtered is a function of the size of the receptive fields. As the size of the receptive field becomes smaller, the resolution becomes higher. In order to take the foveocentric property of the model into account the positions of the receptive fields are defined in a polar coordinate system. The spatial sensitivity profile of the receptive field has the shape of a DOG function. For the combined process of filtering and resampling (explained in Section 1.3.2) the centers of the receptive fields are considered to be distributed on a hexagonal grid. The spacing between the centers of the receptive fields is determined by the high frequency cut-off of the DOG filter. The high frequency cut-off of the DOG filter is inversely proportional to the diameter of the central lobe of the DOG function [Figure 2]. Hence as the size of the receptive fields become larger, the distance between their centers is made longer without introducing any aliasing problem. This distance between the centers of the receptive fields defines the shifting step of the convolution process in Equation (1.7).

1.3.4. SUMMARY OF THE PROPERTIES OF THE PROPOSED IMAGE PROCESSING SYSTEM

**

1. The model is foveocentric.
2. Receptive fields have a spatial sensitivity profile that is similar to a DOG function.
3. There is a ganglion cell associated with each receptive field.
4. The output of a ganglion cell is the result of the convolution of the image

with the spatial profile of the associated receptive field.

5. The size of the receptive fields increases towards the peripheral region of the model, therefore the resolution with which the image is filtered decreases towards the peripheral region of the model.
6. The centers of the receptive fields are distributed on a hexagonal grid.
7. The distance between the centers of the receptive fields increases towards the peripheral region of the model.

In general the system can be referred to as a *combined process of filtering and sampling over an inhomogeneous hexagonal grid*.

1.4. REVIEW OF THE RELEVANT LITERATURE

The idea for a retinal image processing system was first introduced by Koenderink [Koenderink and van Doorn]. In that paper the adaptation property of the ganglion cells were formulated and the effects of this property on the contrast detection were analysed. This work was then continued by Sandini, Braccini [Braccini *et al* 1981 and 1982] and Reitboek. The related physiological data for the development of a model of the retina can be found in papers by Graham, Lennie, Cornsweet, and Drecher.

In Section 1.3.3. the properties of a model of the human retina were pointed out. One of these was the localized processing of images. The same idea for square processing areas was introduced in a paper by Nishihara, where both the localized processing and scale-space techniques were used to develop a binocular-stereo matching algorithm. The scale-space-map and its applications have been addressed by Crowley, Asada, Babaud, Yuille and Witkin.

2. MATHEMATICAL MODELING OF A RETINAL PROCESSING SYSTEM

A new method of image processing was introduced in Section 1.3. The method was based on the structure of the human retina. The properties of a simple model of the human retina were then described in section 1.3.4. It is the purpose of this Chapter to formulate those properties into a linear mathematical form so that the model can be simulated on the computer.

2.1. DESIGN CONSIDERATIONS

Definitions

Some of the parameters of the proposed model are re-named to simplify the description of the design. These are as follows:

1. The ganglion cells are referred to as the *processors*.
2. The spatial area of the central region of a receptive field is referred to as the *processing area*.
3. Referring to Figure 2 (Section 1.2.2), the area covered by the positive values of the DOG function are called the *central-lobe* and the area covered by the negative areas of the DOG function are called the *surround*.
4. The distance of a point from the geometrical center of the image is called the *eccentricity*.
5. The distance between the geometrical centers of the processing areas are named the *resampling distance*, hence the names resampling frequency and resampling points. The above should be differentiated from the raw image sampling distance and sampling frequency.
6. the sample points of the raw image are referred to as *pixels*.
7. The high frequency contents of an image is defined as the *resolution* of

that image. When the resolution is related to a processing area then it defines the high frequency cut-off of the processor associated with that processing area.

Assumptions

1. Position of the centers of the processing areas are calculated in the polar coordinate system.
2. The geometrical center of the image is the origin for the polar coordinate system.
3. The radii of the processing areas increase linearly with eccentricity.
4. The radii of the processing areas are independent of their circumferential position around the origin.
5. The radius of the largest processing area is very much smaller than the dimensions of the image.
6. There is a fixed number of processing areas at each eccentricity around the origin.

Constraints

1. The radius of the smallest processing area must be at least a few pixels. The reason for this constraint will become apparent in the later sections when the impulse response of the processors is considered.
2. The raw image must be completely covered by the processing areas.
3. The raw image is sampled with a rectangular sampling grid. This is because the images were obtained using conventional video cameras.

2.2. DESIGN OF THE PROCESSING GRID

Referring to the assumption (3) and (4) in the previous section, all the processing areas with the same radius lie on the circumference of a circle which is centered on the origin and has a radius of E , where E represents the eccentricity.

In order to satisfy the second constraint (Section 2.1) with the minimum number of processing areas, then the intersection of the neighbouring processing areas must be the vertices of a hexagon as shown in Figure 3.

According to assumption (3), the radii of the processing areas decrease towards the retinal origin. Due to the first constraint described in Section 2.1 there is an eccentricity beyond which the size of the processing areas can no longer be reduced. Therefore assumption (3) is not applicable any more. Hence the image must be partitioned into three distinct regions, the *fovea*, *periphery* and the *semi-periphery*. In the fovea, the processing areas are of minimum radius and independent of eccentricity and orientation. In the periphery the size of the processing areas start from the minimum radius and increase linearly with eccentricity. The purpose of the semi-peripheral region is explained later in this section.

2.2.1. PERIPHERAL REGION

A set of discrete concentric circumferences (centered at the origin) are defined and referred to as the eccentricities E_i . Each of these circumferences is the locus of N equal size processing areas of radii R_i , distributed at equal distances

Figure 3, The spatial arrangement of processing areas for complete coverage of an image with minimum number of processing areas.

around the circumference. Therefore the angle between neighbouring processing areas at a particular eccentricity is:

$$\Theta = 2 \pi/N \quad (2.1)$$

For large values of N and small processing areas, the radii of the processing areas in the consecutive eccentricities can be considered to be the same. Therefore the center of each processing area at the eccentricity E_i forms an isosceles triangle with the centers of the two neighbouring processing areas at the eccentricity E_{i-1} . It should be noted that, since the radii of the processing areas are very much smaller than the associated eccentricity values, the processing areas appear to be distributed on straight lines rather than curved lines that define the circumferential eccentricities (Figure 4). This means that the centers of the processing areas at consecutive eccentricities are shifted by $\Theta/2$.

Processors coordinates

There are N processing areas of radii R at the eccentricity E . The relationship between the radius of each processing area and its eccentricity can be formulated as:

$$2.\pi.E = a.N.R \quad (2.2)$$

where the product " $a.R$ " represents the spacing of the centers of the neighbouring processing areas at the same eccentricity E (Figure 4). The limits of the variable " a " depend on the geometrical distribution of the processing areas. This is explained as follows: There are two ways to arrange the processing areas to satisfy the second constraint mentioned in Section 2.1. One of these is shown in Figure 4 where there are no overlaps between the

Figure 4, The position of the processing areas with respect to the associated eccentricities.

neighbouring processing areas at the same eccentricity (eg; E_{i+1}). The other arrangement is achieved by rotating the arrangement of Figure 4 by 60 degrees around the center of the processing area at E_i (shown in Figure 5) in which there are overlaps between the neighbouring processing areas at the same eccentricity (eg; E_{i+1}). From Figure 5, it can be seen that the value of the variable " a " is fixed to approximately $\sqrt{3}$. Whereas with the arrangement of Figure 4, " a " can take any value larger than 2. If " a " is lower than or equal to 2 then the intersection of the neighbouring processing areas will not form a hexagon. The upper limit of the value of " a " is determined by the dimensions of the image. The arrangement of Figure 4 is a better choice since it introduces some degree of flexibility in the design of the processing grid.

The third assumption (Section 2.1) describes the size dependence of the processing areas on eccentricity, which can be shown as

$$R_i = K E_i \quad (2.3)$$

where K is the constant of proportionality and is deduced from Equation 2.2 to be:

$$K = 2 \pi / (a N) \quad (2.4)$$

The distance between the centers of the neighbouring processing areas was defined as the resampling distance. Assuming that locally the radii of the neighbouring processing areas are equal, then referring to Figure 6 the resampling distance is given by:

Figure 5, The position of the processing areas with respect to the associated eccentricities, when the arrangement of Fig. 4 is rotated by 60 degrees around the center of the processing area at E_i .

$$d_i = R_i \sqrt{3} \quad (2.5)$$

In Figure 6, ϕ represents the angular shift of the centers of the processing areas at the consecutive eccentricities and is given by:

$$\phi = \theta/2 = \pi/N \quad (2.6)$$

In the triangle OAH,

$$(d_i)^2 = (E_i)^2 + (E_{i+1})^2 - 2E_i E_{i+1} \cos(\phi) \quad (2.7)$$

substituting the expression in 2.5 for d_i and the expression in 2.3 for R_i and then re-arranging to have a quadratic equation in terms of E_{i+1} :

$$(E_{i+1})^2 - E_{i+1}(2E_i \cos(\phi)) + (E_i^2 - 3K^2 E_i^2) = 0 \quad (2.8)$$

solving for E_{i+1} ,

$$E_{i+1} = \lambda E_i \quad (2.9)$$

where the constant λ is a function of the number of equal size processing areas and is given by

$$\lambda = \cos(\phi) + \sqrt{(3K^2 + \cos^2(\phi) - 1)} \quad (2.10)$$

The number of discrete eccentricities before reaching the edges of the image is proportional to λ . This can be shown as follows: According to Equation 2.3, the radii of the processing areas decrease towards the origin. The radius of the smallest processing area however, must be a few pixels (Section 2.1). Taking this

Figure 6, The spacing of the neighbouring processing areas with respect to their radii and the associated eccentricities.

radius as R_0 and the minimum eccentricity in the peripheral region as E_0 , then the consecutive eccentricities and the associated radii of the processing areas can be formulated in terms of E_0 and R_0 . Using Equations 2.3 and 2.9,

$$\begin{aligned} E_1 &= \lambda^1 E_0 \text{ and } R_1 = \lambda^1 R_0 \\ E_2 &= \lambda^2 E_0 \text{ and } R_2 = \lambda^2 R_0 \\ E_I &= \lambda^I E_0 \text{ and } R_I = \lambda^I R_0 \end{aligned} \quad (2.11)$$

Where I represents the total number of eccentricities before reaching the edges of the image. Taking G as half the original image dimension, then the following inequality must hold.

$$E_I + \gamma R_I \leq G \quad (2.12)$$

γ is incorporated in order to include the inhibitory region of the DOG function as well as the excitatory region which is referred to as the processing area. Therefore γR_I represents the radius of the whole DOG function. Substituting Equation 2.11 into 2.12,

$$\begin{aligned} \lambda^I E_0 + \gamma \lambda^I R_0 &\leq G \\ I &\leq \text{Log}(G/(E_0 + \gamma R_0)) / \text{Log}(\lambda) \end{aligned} \quad (2.13)$$

It is clear from Equations 2.3, 2.4, 2.6, 2.9 and 2.13 that the coordinates of the centers of the processing areas in the peripheral region are specified by the parameters N , R_0 and a . R_0 is constrained by the impulse response of the processors. The values of a and N are application dependent. According to Equation 2.4, both N and a determine the value of K . Therefore for a given value of R_0 , using Equations 2.3 and 2.4, N and a determine the value of E_0

which defines the extent of the foveal region. Also for a given value of R_0 , using Equations 2.4, 2.6 and 2.10, N and α determine the value of λ which in turn determines the number of discrete eccentricities in the peripheral region (Equation 2.13). Therefore the values of N and α determine the smoothness with which the resolution is reduced in the peripheral region. It should be noted that E_0 can replace α as an input parameter without reducing the design flexibilities.

Design algorithm

The following procedure can be used to construct the peripheral structure:

1. The parameters R_0 , N and α are chosen according to the application. The values of the parameters K , E_0 , ϕ and λ are then calculated using Equations 2.4, 2.3, 2.6 and 2.10 respectively.
2. The circumference defined by E_0 is then partitioned into N equally spaced points.
3. The next allowable eccentricity is evaluated, using equation 2.9.
4. The starting point at this eccentricity is then shifted by ϕ .
5. As long as the inequality 2.12 is not violated, repeat steps (2) to (4).

The above points correspond to the geometrical centers of the processing areas. The size of the processing areas at each eccentricity is then determined by Equation 2.3.

2.2.2. FOVEAL REGION

It was desired to have a high resolution processing region that was compatible with conventional vision systems so that the results of the processing of images in this region could be directly linked to other processing modules that already existed. Based on the above requirements, the foveal region must consist of

minimum size processing areas at every sample point of the raw image. Hence the resampling distance in the foveal region is equal to the sampling distance of the image, whereas the resampling distance immediately outside the fovea in the peripheral region is given by Equation 2.5. Therefore there is a discontinuity in the resampling distance at the fovea/peripheral boundary. One solution for the above problem is to introduce another processing region between the fovea and the the periphery, namely the semi-peripheral region.

2.2.3. SEMI-PERIPHERAL REGION

The semi-peripheral region is merely a continuation of the periphery. The radii of the processing areas in this region are constant but the resampling distance d_i is still a function of the eccentricity. This means that Equation 2.9 still holds, while Equation 2.3 is discarded. Starting from the minimum eccentricity E_0 in the peripheral region and rearranging Equation 2.9 to

$$E_{i-1} = E_i / \lambda$$

then the same procedure as in the peripheral case can be employed to construct the semi-peripheral region. Although here, step 1 is discarded and step 5 of the algorithm is replaced by :

5. as long as the inequalities

$$d_i \geq 1, \text{ or}$$

$$E_i(1 - 1/\lambda) \geq 1$$

are satisfied, repeat steps (2) to (4).

2.3. PROCESSING OF IMAGES

A processor was assumed to be associated with each of the processing areas described in the previous section. The function of the processors is to extract useful information from the images. The spatial profile of each processing area defines the impulse response of the corresponding processor.

2.3.1. IMPULSE RESPONSE OF THE PROCESSORS

Referring to Section 1.2.2, processing must result in the enhancement of the intensity changes and the suppression of uniform intensity profiles in the image. Also the resolution with which the image is processed must depend on a controllable parameter, so that it can be incorporated in the model described in Section 1.3.3. The *operators* that satisfy the above requirements are considered next.

Laplacian of a Gaussian

One of the operators that fulfills the above requirements, is the Laplacian of a Gaussian operator given by:

$$\nabla^2 G(r) = (1-r^2/2\sigma^2) \cdot \text{Exp}[-r^2/2\sigma^2]/(\pi\sigma^2) \quad (2.14)$$

with its spatial frequency response, given by

$$F(\omega) = 4\pi^2\omega^2 \cdot \text{Exp}[-2\pi\omega^2\sigma^2] \quad (2.15)$$

It is clear from Equation 2.15 that this operator has a band-pass characteristic. Its center frequency and bandwidth are functions of the space constant σ . Lunscher [Lunscher and Beddoes,(I) 1985] has shown that the minimum edge spacing detectable by this operator, for an ideal step change in the intensity, is

$$T = 2.75\sigma \quad (2.16)$$

and for a blur† of variance 0.5σ , is

$$T = 5.5\sigma \quad (2.17)$$

It is clear from equation 2.14 that the radius of the central lobe of this operator is given by

$$R = \sigma\sqrt{2} \quad (2.18)$$

With the exception of optical signal processing, image processing involves sampled data. Such discrete signal processing requires the filter to be sampled and quantized. It is clear from Equation 2.15 that the filter response is not band-limited and the sampling process causes a certain degree of aliasing. Therefore σ can not be reduced in value, indefinitely. The minimum value of σ is a function of the sampling distance "s" in the image plane. Lunscher has evaluated this to be:

$$\sigma_{min} = 0.8s \quad (2.19)$$

and has shown that at this sampling rate, 2.7% of the filter energy lies above the sampling frequency. Any σ of magnitude less than this value is not practical, due to the excessive aliasing energy. According to Equation 2.18 and 2.19, the central lobe of the smallest operator should at least cover 3 sample points along its diameter. The extent of the *surround* however, is a function of the quantization process. After quantizing the filter coefficients, the most notable

† The blur function is considered to be a Gaussian function of zero mean

effect is the appearance of a strong DC-offset, which reduces the accuracy of the processing as far as the position of an edge is concerned. Lunscher points out that this DC-offset can be removed by first summing all the quantized coefficients and then subtracting the result of the summation from the coefficients [Lunscher and Beddoes (II), 1985].

Difference of two Gaussians

It can be shown that the difference of two Gaussians (DOG) operator is a good approximation to the $\nabla^2 G$, when the ratio of the space constants of the Gaussians is about 1:1.6 [Marr and Hildreth 1980]. The mathematical expression for the DOG function is given by:

$$\text{DOG}(r) = \frac{A}{\sigma_e \sqrt{2\pi}} \text{Exp}[-r^2/2\sigma_e^2] - \frac{1}{\sigma_i \sqrt{2\pi}} \text{Exp}[-r^2/2\sigma_i^2] \quad (2.20)$$

where σ_e represents the space constant of the *Excitatory* Gaussian, and σ_i represents the space constant for the *Inhibitory* Gaussian (section 1.2.2, Figure 2). The function of the parameter "A" will be discussed later.

As it was mentioned earlier, the DOG function is a good approximation to $\nabla^2 G$ when the following equality holds,

$$\sigma_i = 1.6 \sigma_e \quad (2.21)$$

If the radius of the central lobe of the DOG function is represented by R, then

$$\text{DOG}(R) = 0 \quad (2.22)$$

Substituting σ for the space constant of the *Excitatory* Gaussian and using Equations 2.20, 2.21 and 2.22, then

$$\begin{aligned}
 A \cdot \text{Exp}[-R^2/2\sigma^2] &= 0.625 \text{ Exp}[-R^2/5.12\sigma^2] \\
 \text{Ln}(1.6A) &= 1.56 R^2/5.12\sigma^2
 \end{aligned}
 \tag{2.23}$$

It is clear from Equation 2.20, that the DOG function is not space limited and that for digital implementation, some form of windowing is required to limit the spatial extent of the profile. This however, as in the case of the $\nabla^2 G$, will result in some DC-offset. In the case of the DOG function, it is clear from Equation 2.23 that the radius R of the central lobe of the operator is a function of two parameters A and σ . By adjusting the value of A , keeping the radius R and the space constant ratio σ_i/σ_e constant, the DC-offset can be reduced to zero.

2.3.2. PROCESSING ALGORITHM

The impulse response of the processing areas was chosen to be the DOG function. This was simply because of Equation 2.23 (Section 2.3.1) which provides an easy way of suppressing the DC-offset. The processing performed by each unit can be represented by the following equation:

$$f(X,Y) = \sum \sum I(\xi,\mu) \text{DOG}(X-\xi, Y-\mu) \tag{2.24}$$

where $I(X,Y)$ is the input image and $f(X,Y)$ is the output. The spatial profile of the DOG function is circularly symmetric. Therefore its value at each point is determined by the distance of that point from the geometrical center of the function. It is therefore convenient to define the coefficients of this function in a Polar coordinate system. This is a local coordinate system with its origin at the

geometrical center of the DOG function.[†] Referring to Section 2.2.1, the geometrical center of the image is the origin for a global polar coordinate system in which the coordinates of the centers of the processing areas are defined. Each of these centers are in turn the origin for a local polar coordinate system in which the values of the coefficients of the DOG function are defined.

Bearing in mind that a processing area corresponds to the central lobe of a DOG function, then according to Equation 2.3 (Section 2.2.1) the size of the central lobe of the DOG function increases with eccentricity. Taking into account the circular symmetry property of the DOG function and its size dependence on the eccentricity, then Equation 2.24 can be written in a polar form,

$$f(E, \theta) = \sum_r \sum_{\phi} r \cdot \text{DOG}(r/E, E \cdot \cos \theta - r \cdot \cos \phi, E \cdot \sin \theta - r \cdot \sin \phi) \quad (2.25)$$

where E and θ represent the global co-ordinate system and r and ϕ represent the local co-ordinate system. According to Equation 2.25, all the image sample points at a radius r from the center of the processing area are added together and then multiplied to the coefficient of the DOG function at that radius.

Figure 7 shows the function $\text{DOG}(r, \sigma)$, where r represents the radius of a point in the function and σ represents the space constant of the Gaussian. The spatial area of the DOG function is partitioned into incremental stripes, each with a width of

[†] The geometrical center of the DOG function corresponds to the geometrical center of the associated processing area.

Figure 7, The spatial sensitivity profile of a DOG function and the associated stripes for local coordinate transfer.

Δ_i , equal to the sampling distance on the original image. If all the image sample points within each strip are considered to be at the same distance from the center of the DOG function, then the rectangular distribution of the image samples can be rearranged into this approximately circular distribution. Therefore to implement Equation 2.25, all the image points that fall within each strip are added together and then multiplied by the weighting factor associated with that strip, namely $\text{DOG}(r_i)$. If the radius of the DOG function is taken to be R , then the following algorithm can be used to construct these strips on a rectangular distribution of sample points.

1. Local origin is set to the center of the processing area.
2. The X-axis is chosen to be the reference for the values of the distance r from the local origin.
3. Starting from the origin ($r = 0$).
4. While ($0 \leq x \leq r$) do step 5.
5. Check for values of y that satisfy the inequality

$$r - 0.5 < \sqrt{(y^2 + x^2)} < r + 0.5$$

6. If ($r < R$), then increment r by one and go to step 4.

In step 5, all the sample points that are within ± 0.5 of a circumference of radius r from the origin are grouped together and stored. It must be notice that the co-ordinates of all the sample points within each strip are symmetrical with respect to the local origin. Therefore the inequality of step 5 must only be checked for the first quadrant. The coordinates of the rest of the sample points within each strip in the other three quadrants are the combination of $\pm x$ and $\pm y$.

3. EVALUATION OF A RETINAL PROCESSING SYSTEM

3.1. DESCRIPTION OF THE PROCESSING GRID

The processing grid was simulated on the VAX-11/750 and the coding was completed in FORTRAN. The program required three inputs

1. The dimension of the image (eg; ...128,256,512...) which was referred to as 2G.
2. The radius of the smallest processing area R_0 .
3. The number N of equal size processing areas at any eccentricity in the peripheral region.

The spacing $a.R$ between the centers of the processing areas at any eccentricity E_i was chosen to be $3R$ ($a=3$), where R is the radius of the processing area at that eccentricity. The reason for this value of a is as follows:

1. a was chosen to be an integer rather than a real number, to reduce the computational time, and
2. a was chosen to be the smallest allowable integer ($ie > 2$) so that the effect of small values of N on the smoothness of the resolution change in the peripheral region could be examined (Section 2.2.1).

It is however possible to choose values of a much greater than 3. The number N , of equal size processing areas at each eccentricity was chosen to be 64. There is no particular reason for this value of N . It is however suggested to choose the value of N to be a power of 2 (ie ; ...32, 64, 128...). This becomes useful when considering the development of the hardware for the proposed retinal image processing system (see Section 4.1). The radius of the central lobe of the smallest processing area was 2 pixels (Section 2.3.1) and the image dimensions

were 256x256 pixels. Using the above values of α , N and R_0 , the rate of increase of the size of the processing areas with eccentricity was then evaluated (Equation 2.4 Section 2.2.1)

$$K = 0.03273$$

Using Equations 2.6 and 2.10, the constant λ , corresponding to the ratio of the consecutive eccentricities was then computed

$$\lambda = 1.02717$$

which means that each processor at eccentricity E_i is 2.72% larger than the processor at the previous eccentricity. The minimum eccentricity in the peripheral region corresponding to the smallest size processing area was then calculated using Equation 2.3 (section 2.2.1). It should be noted that if the inequality 2.12 is not satisfied for the minimum eccentricity (ie; $E_I = E_0$), then the system would send a warning message to the user to either reduce the number of the equal size processing areas N , or to use an image with larger dimensions.† If the inequality 2.12 for E_0 was satisfied, then the following was carried out to construct the processing grid.

1. The number of discrete eccentricities "I", in the peripheral region was computed (Equation 2.13)

$$I = 24$$

2. A two dimensional array $A[I \times N]$ was then constructed; where "I" was the number of discrete eccentricities and N was the number of equal size processing areas at each eccentricity (This stage is particularly useful for hardware implementation, as to how much storage is required for the

† The dimensions of the input image are normally fixed, and depend on the camera specifications. It is however possible to enlarge the image, using Zero-padding in the Fourier domain.

processing grid). $A[I \times N]$ was used as an addressing matrix, whereby random access to the co-ordinates of the processors was achieved. Each element $a_{i,n}$ was a $[2 \times 2]$ matrix itself, containing both the polar and the cartesian co-ordinates of a particular processor.

$$a_{i,n} = \begin{bmatrix} r & \theta \\ x & y \end{bmatrix}$$

$$r = E_i \quad i = 0, 1, 2, \dots, I-1$$

$$\theta = 2n\pi/N \quad n = 0, 1, 2, \dots, N-1$$

$$x = E_i \cos(2n\pi/N)$$

$$y = E_i \sin(2n\pi/N)$$

The reason for constructing such matrices will become clear in the following sections of this chapter.

3. The number of allowable eccentricities in the semi-peripheral region was computed using the constraint imposed by the following inequality

$$E_i - E_{i-1} \geq 1$$

The minimum eccentricity in the semi-peripheral region was related to the minimum eccentricity in the peripheral region (using Equation 2.11 and replacing λ with $1/\lambda$), by:

$$E_{-F} = \lambda^{-F} E_0$$

where E_0 represented the minimum eccentricity in the peripheral region and E_{-F} represented the minimum eccentricity in the semi-peripheral region and F was the number of allowable eccentricities in the semi-peripheral region (the negative sign is merely for stressing the fact that the eccentricities are getting smaller). F and E_{-F} were then evaluated using the equations:

$$E_{-F} = (\lambda-1)^{-1} = 37$$

$$F = \text{Log}(E_0(\lambda-1))/\text{Log}(\lambda) = 19$$

4. Step (2) was repeated with "I" replaced by "-F".
5. In the foveal region, a processor was considered to exist at every sample point within a circumference of radius E_{-F} .

Figure 8 shows the processing grid produced by the above algorithm.

Once the grid for a particular set of parameters (*ie*; N, G and R_{min}) was designed, the coordinate information describing that grid, was stored and used as a look-up table for the higher processing stages such as filtering, edge detection, boundary following *etc.*

3.2. THE RESULTS OF BAND-PASS FILTERING

The radius of the largest processing area was used to perform local co-ordinate transfer (Section 2.3.2). The radius of the DOG function (taking into account, the extent of the inhibitory region) was chosen to be twice the radius of the central lobe. The reason for this is that the coefficient of the DOG function at $2R$ (where R represents the radius of the central lobe) is about 0.3% of the largest coefficient. Therefore all the coefficients at radii greater than $2R$ can be ignored. The incremental radii used for the local coordinate transfer, were chosen to be one pixel apart. This simplified the grouping of the sample points (Section 2.3.2). The largest processing area had a radius of 4 pixels. Hence the radius the DOG function associated with that processing area (taking into account, the inhibitory region) was $2 \times 4 = 8$ pixels. This meant that for the largest processing area the local co-ordinate transfer resulted in 8 concentric strips. The constant "A" of the

Figure 8, The distribution of the centers of the processing areas.

DOG function (Section 2.3.1) was adjusted by iterative methods so that the DC-offset of the processor was less than 0.1% of the intensity values. Thus for uniform intensities of maximum gray level of 256, the DC-offset was only about 0.2, which is less than the gray level intervals. The image sample points within each strip of the processing area were added together and then multiplied by the corresponding filter coefficient (*i.e.*; the value of the DOG function at the average radius of the strip from the center of the processing area). The procedure was repeated for every allowable point on the designed grid in the peripheral, the semi-peripheral and the foveal region.

As it can be seen from Figure 1 (Section 1.2), convolution of the DOG with a step change in the intensity results in negative responses. In order to be able to show the results of filtering on the monitor, all the values must be shifted by some constant positive value to prevent any negative intensity. The results of filtering were bounded between ± 128 , hence a d.c. intensity of +128 was added to the results of filtering. This addition however, is not necessary if the filtered image is directly used for higher processing (such as zero-crossing detection) without the need to show the intermediate results.

Figure 9, shows the unprocessed image (256x256x8 bits), and Figure 10 shows the results of filtering this image based on the grid of Figure 8.

3.3. EDGE DETECTION IN THE IMAGE

From Figure 1 (Section 1.2), it can be seen that for a step change in the intensity the filter produces a zero-crossing at the point of the intensity change. As previously mentioned, the intensity values in the filtered image were shifted

Figure 9, The raw image on a rectangular sampling grid (256x256x8 bits)

Figure 10, The result of the filtering of the image in Fig. 9, based on the processing grid shown in Fig. 8.

by a positive DC value to prevent the occurrence of negative intensities. This shifting procedure meant that the zero-crossings were transformed to *Level-crossings*, where the reference level was equal to the DC value. This DC value was chosen to be +128. Hence, if the neighbouring pixels in the filtered image had values greater and smaller than 128 respectively, then a level-crossing was assigned which corresponded to an edge in the unprocessed image. The strength and the direction of the level-crossings in the filtered image were stored as information describing the edges in the unprocessed image.

It was explained in Section 3.1 that each element $a_{i,n}$ in the matrix $A[I,N]$, was a 2×2 matrix containing both the polar and the Cartesian co-ordinates of the designed resampling grid. The Polar coordinates were used to find the level-crossings in the filtered image. In order to check for the existence of a level-crossing at $a_{i,n}$ in the filtered image, the following steps were carried out,

1. if the intensity value $I_{a_{i,n}}$ at $a_{i,n}$ was greater than 128, then
 - a. the intensity values of the neighbouring pixels $(i-2,n)$, $(i-1,n-1)$, $(i+1,n-1)$, $(i+2,n)$, $(i+1,n+1)$ and $(i-1,n+1)$ were checked, bearing in mind that $a_{i,n}$ was completely isolated by these six *target pixels*.
 - b. If an intensity of less than 128 was met, then a level-crossing was assigned to $a_{i,n}$ with its amplitude as $(I_{a_{i,n}} - I_b)$, where "I" represents the intensity value and b represents the co-ordinates of the target pixel. A local direction was then assigned to the level-crossing according to the co-ordinates of the target pixel. Table 1 shows the corresponding local directions for each target pixel.
 - c. if an intensity of 128 was met, then the next pixel in the same

Target pixel	local direction
$i-2,n$	0
$i-1,n-1$	$\pi/3$
$i+1,n-1$	$2\pi/3$
$i+2,n$	π
$i+1,n+1$	$4\pi/3$
$i-1,n+1$	$5\pi/3$

Table 1, The local direction of the target pixels with respect to
the center pixel $[i,n]$

direction was checked and if its intensity value was less than 128, then a level-crossing was assigned to the pixel with the intensity of 128. If however, the next pixel had an intensity value of greater than 128 then the candidate in the above direction was ignored.

2. if the intensity value at $a_{i,n}$ was less than 128 then the whole process of step(1) was repeated, with the difference that the target pixel had to have an intensity value of greater than 128.
3. if two or more level-crossings were detected for $a_{i,n}$, then the level-crossing with the largest amplitude was chosen to represent the edge.

When a pixel with an intensity value of 128 was detected, then a number of neighbouring pixels in one particular direction were checked (step(1.c) above). It should be noted that the number of pixels that are checked in this manner, can be used as a thresholding method for ignoring the very weak edges in the image. For example if at $a_{i,n}$ the intensity is greater than 128, and at $(i-2,n)$ the intensity is 128, then the intensity of the next pixel $(i-4,n)$ which lies in the

same local direction would be checked. If at $(i-4,n)$ the intensity is

1. greater than 128, then no level-crossing is assigned
2. less than 128, then a level-crossing is assigned to the pixel at $(i-2,n)$
3. equal to 128, then the following is applicable
 - a. the pixel at $(i-2,n)$ is ignored as a candidate for level-crossing
 - b. the pixel at $(i-6,n)$ is checked for intensity of less than 128

In case 3.a, the level-crossings of slopes less than $1/2$ (gray-level/pixel) are ignored whereas in case 3.b, these level-crossings are still considered. This is why the number of pixels that are checked in one particular direction behave as a thresholding method for detection of edges based on their amplitudes. In this project, only 2 pixels were checked at a time, which means that the level-crossings of slopes less than 1 gray-level/pixel were ignored.

Figure 11 shows the edges detected in the filtered image of Figure 10. The brightness of the pixels in the image is proportional to the amplitudes of the edges. It should be noted that some of the edges do not appear on the screen due to their small amplitudes. This can be verified by binerizing the filtered image in Figure 11 to a threshold of 2, so that most level-crossings appear as amplitudes of 255 (Figure 12).

3.4. EDGE FOLLOWING IN THE FILTERED IMAGE

Figure 12 shows a set of edges that were obtained by detecting the level-crossings in the filtered image. In order to extract the connected edges and construct *boundaries*, certain constraints must be satisfied. These are:

1. at least "M" number of edges must exist in a boundary segment,

Figure 11, The detected edges in the filtered image shown in Fig. 10.

Figure 12, The binary equivalence of the image shown in Fig. 11, where all the edges of amplitude greater than 2 were set to amplitudes of 255

2. the local orientation difference of the neighbouring edges must be limited to a certain angle values (eg; $2\pi/3$).

The value of the integer "M" is optional and can be as small as 1. Small values of M however, reduces the noise tolerance of the system. The value of M was arbitrarily chosen to be 3 (ie; all the segments with only 1 or 2 members were ignored as candidates for boundaries). The second constraint ensures the local smoothness of the boundaries.

There are certain properties of the proposed model that can be used to simplify the process of extracting boundaries from the images. One of these properties is that, each processing area in the model is completely isolated by its six neighbouring processing areas from the rest of the processing areas. This property makes *Dynamic Programming* a suitable choice for grouping the edges as segments of boundaries. The basic idea is to construct a cost function f_i that takes into account the strength of the edge i and its orientation difference with the previous candidate edge $i-1$. This can be shown as follows;

$$f_i(e_i, e_{i+1}) = A(e_i) - \mu \cdot q(e_i, e_{i+1}) + f_{i-1}$$

where:

e_i represents the i^{th} edge in the image.

A represents the amplitude of an edge

q is the magnitude (Mod. π) of the difference between the local orientation of two neighbouring edges.

μ is a weighting factor (or function).

The function f should be maximized over all its candidate edges. μ can be chosen according to any particular application. Its role in the cost function f is

to emphasize the effect of the local orientation of the edges. For example if μ is an exponential function of q , then the edges that lie on a straight line are most likely to be grouped together as segments of a boundary. μ was chosen to be a constant of value 81. This is because the strongest edge can have an amplitude of 255 with its orientation ranging from zero to $5\pi/3$; If the neighbouring edges are in opposite direction (ie, a local orientation difference of π), then $\mu\pi \approx 255$, which cancels the effect of the amplitude of the edge in the cost function, hence acting as if the edge did not exist.

Each group of edges has an initial node e_1 and an end node e_i , with i members and a cost function f_i . Based on the above definition the following algorithm was used to group the edges together as segments of boundaries and then grouping these segments to obtain a connected boundary of an object.

1. Using the results of the previous stage (Section 3.3), the edge with the largest amplitude was chosen to be the starting point.
2. A table was then constructed for the boundary segments. Each segment was labeled $P(k)$ by which it could be accessed in the table, where initially

$$P(k) = k \quad (k=1,2,3,\dots)$$

The following information was extracted once the segment was accessed through its label;

- a. The co-ordinates of the initial node, $PI(k)$
 - b. The co-ordinates of the final node, $PF(k)$
 - c. Number of the edges in the segment L ,
 - d. The cost of constructing such a segment, $f(k)$
3. Every edge in the image was initially labeled $e(i,n)=0$, which meant that it

did not belong to any segment. "i" and "n" represented the polar co-ordinates of the edge on the grid.

4. "k" was set to 1 in the table of step(2)
5. Starting from the edge with the largest amplitude, its coordinates were assigned to $PI(k)$, and the edge was labeled $e(i,n)=k$, indicating that it belonged to the segment k.
6. The three neighbouring pixels in a direction perpendicular to the local orientation of the edge at $PI(k)$ were checked for the existence of an edge;
 - a. If there was only one unlabeled edge, then its local orientation was used for calculating the cost of taking this candidate as part of the segment. Its co-ordinates were then assigned to $s(i,n)$, where i and n are the polar co-ordinates of the previous edge. It should be noted that this method of addressing is generally referred to as the *Chain rule*, where each parameter points to the next one in line.
 - b. If there were two or three unlabeled neighbouring edges then the co-ordinates of each edge is assigned to $s1$, $s2$ and $s3(k)$. Replacing k with $k+1$, another parameter $BT(k)$ was then introduced, indicating the need for *Back Tracking*, once the new segments were completed.

$$BT(k) = k - 1$$

The reason for $BT(k)$ was that each of these new segments were separately completed and their cost were compared for choosing the best segment as the continuation of the previous segment.

7. Starting from $s1(k)$, steps (5) and (6) were repeated until all the unlabeled neighbouring edges were checked.
8. $BT(k)$ was then checked and if not equal to zero, then step(7) was

repeated for $s_2(BT(k))$, $s_3(BT(K))$.

9. Once the boundary segments for all s_1, s_2 and s_3 were completed, their cost functions were compared and the largest was chosen as the continuation of the previous segment, $P(BT(k))$.
10. All the edges $e(i,n)=k$, where then relabeled $e(i,n)=BT(k)$.
11. steps (5) to (10) were repeated until there were no unlabeled edges in the image.
12. finally the initial and the final nodes of all segments were compared to see if any two segments had their end nodes as neighbours. The reason for this step is that, once a segment is completed, then its members can not be a member of any other segment even if the new segment ends in the neighbourhood of some other segment. If any two segments had their end nodes as neighbours, then they were fused to one segment.

It should be noted that the segments with the largest cost or the highest number of members are candidates for the boundary of an object in the image.

Figure 13 shows the result of applying this algorithm to the image of Figure 11.

3.5. BOUNDARY FOLLOWING IN THE PROCESSED IMAGE

Here, the proposed grid is referred to as the *Eye*, due to its resemblance to the retina in the human visual system. As it was explained in Chapter 1, the function of the periphery is to provide low resolution information about a scene while the fovea is responsible for detail analysis of that scene. Once the required feature is extracted from the peripheral region, the fovea can be moved to the approximately position of that feature to obtain further information. The

Figure 13, Results of applying dynamic programming to the image shown in Fig. 12, for grouping of edges to form boundary segments.

advantage of this method of analysis is the reduction in the number of computations, as explained below.

A scale-space-map is based on the multiresolution analysis of an image at every point on the image [J. Babaud *et al*, A.P. Witkin, J.L. Crowley, A.L. Yuille and T.A. Poggio]. The proposed model however, behaves as scale-space-map of edge-lines in the filtered image, that pass through the fovea. This is because of the smooth reduction in the resolution with which the image is processed towards the far peripheral region. In the conventional scale-space-map, a two dimensional map (σ, x) is employed to analyse a one dimensional slice of a two dimensional image. In the proposed model, a three dimensional map (σ, x, y) is employed to analyse the two dimensional image. In the latter, the degrees of freedom is increased by one, but the system is constrained to the analysis of lines. If a boundary is extended from the fovea to the peripheral region, then starting from the peripheral end of the boundary (low resolution), it is followed through the semi-peripheral region into the foveal area (high resolution). Once the true boundary segment in the foveal region is detected, then the Eye is moved in the direction of the boundary, towards that section of the boundary which lies in the low resolution area. The processing in the foveal region after the Eye movement is restricted to a small region along the path of the boundary which was detected in the previous frame of the periphery. This Eye-movement is continued until a discontinuity is detected in the boundary or if the boundary closes on itself. It is obvious that most of the computational burden is due to the processing of image in the high resolution area (*ie*, the fovea). The above method of analysis reduces the computational cost, by processing only certain areas of

the fovea.

The Eye movement was simulated on the VAX-11/750, for the image of Figure 13. The starting point was chosen to be in the peripheral region. The initial node of the boundary segment with the highest cost function was taken to be the starting point. This point turned out to be in the top portion of the hat of the girl. The raw image was then shifted so that the coordinates of the initial node of the boundary segment was at the center of the foveal region. The raw image in the fovea was filtered only in a region of 10 pixels wide along the path of the boundary segment. Note that all the above information (initial node, path and the final node of the boundary segment) were all available from the previous stage of processing, described in Section 3.4. The raw image in the peripheral and semi-peripheral region was not restricted to any small section of processing and was filtered over the whole region. The procedure described in Sections 3.2, 3.3 and 3.4 was then repeated. Any boundary segment in the semi-peripheral region that was in the neighbourhood of the segment in the foveal region was next brought into the foveal region. Figure 14 shows the result of the Eye-movements. The section of the hat in the image of Figure 14 was obtained by 10 Eye movements.

3.6. COMPUTATIONAL COST OF RETINAL PROCESSING

Fovea

The foveal region has a diameter of 74 pixels, therefore the number of sample points in the foveal region is approximately equal to 4300. Since there is a processor at every sample point, then the number of processors in the foveal

Figure 14, Results of boundary following algorithms for the detection of the boundary of the hat of the girl shown in Fig. 9.

region is also equal to 4300. The processing areas associated with these processors are of minimum radius (Section 2.2.2). Hence, by the convention of Section 3.3, the radius of the whole processing area (taking into account the extent of the inhibitory area and the sample point at the center of the processing area) is 5 pixels. Therefore, taking advantage of the circularly-symmetric property of the DOG function The total number of multiplications for each processing area is 5 (Section 3.3). The total number of the multiplications in the foveal region is therefore given by

$$4300 \times 5 = 21,500$$

Semi-periphery

In the semi-peripheral region there are 19 allowable eccentricities, each with 64 processing areas of minimum radius. Hence using the same argument as the foveal case, the total number of multiplications is given by:

$$19 \times 64 \times 5 = 6,080$$

Periphery

In the peripheral region there are 24 allowable eccentricities, each with 64 processing areas. But the radii of the processing areas are not constant and vary linearly with eccentricity. Taking advantage of this linearity, the average radius of the processing areas can be used to estimate the total number of multiplications. The largest processing area in the periphery has a radius of 4 pixels and the smallest has a radius of 2 pixels. Therefore the average radius of the processing areas is 3 pixels. Following the same procedure as in the case of the fovea and the semi-periphery, then the total number of multiplications in the peripheral region is given by:

$$24 \times 64 \times 3 = 10,752$$

Hence the total number of multiplications, including the fovea and semi-periphery is 38,332. The number of multiplications in extracting the boundary of the hat of the girl, using the procedure of section 3.5, can be calculated as follows; The image of Figure 14 was obtained by 10 eye-movements. It is evident that this number could be reduced by increasing the size of the fovea. For each eye-movement the complete computation of Z.C.s was carried out in the peripheral and semi-peripheral regions. Using the above values, the total number of multiplications in the peripheral and semi-peripheral region for 10 eye-movements is given by:

$$10 \times (10,752 + 6,080) = 168,320$$

In the foveal region however, the complete computation of Z.C.s was carried out only once and that was for the very first frame. In the following 9 frames only parts of the foveal region were processed. These regions were on average 20 samples long and 10 samples wide. Hence, each area contained 200 sample points and therefore 200 processing area of radii 5. The total number of multiplications for the 10 frames analysis, is given by:

$$21,500 + (9 \times (200 \times 5)) = 30,500$$

The total number of multiplications in extracting the hat is approximately given by:

$$168,320 + 30,500 = 198,820$$

The number of multiplications in filtering the same image, based on rectangular sampling and with smallest size processing areas (highest resolution) of radii 5, is given by:

$$256 \times 256 \times 5 = 327,680$$

Evidently, the proposed model results in

$$(327,680 - 198,820)/327,680 = 39.3\%$$

reduction in the number of multiplications. As it was mentioned before, the proposed model has taken advantage of scale-space techniques which has not been considered in evaluating the above number of multiplications for conventional vision systems. If the utilization of scale-space techniques is also considered for the conventional scheme, with only 3 levels of filtering (*ie*, radii of 5, 7 and 9) which is equal to the number of resolution levels in the periphery, then the total number of multiplications for the conventional scheme is given by:

$$256 \times 256 \times (5 + 7 + 9) = 1,376,256$$

Comparing this with the number of multiplications for the proposed model, it can be seen that the proposed model results in

$$(1,376,256 - 198,820)/1,376,256 = 85.6\%$$

reduction in the number of multiplications.

3.7. ERROR ANALYSIS

Processing grid

In constructing the processing grid, the allowable eccentricities and the position of the processors around the circumferences of radii equal to these eccentricities, were evaluated according to the equations presented in Chapter 2. These equations were developed based on the assumption that both the spacing of the eccentricities and the radii of processing areas were locally constant. As a result, the spacing of the processing areas (resampling distance) in the peripheral region was shown to be $\sqrt{3}R$ (Section 2.2.1), where R was the radius of the processing area. The validity of the above assumption and the resulting relationship between the resampling distance and the radius of the processing areas can be shown as

follows; The consecutive eccentricities in the peripheral region are given by

$$E_{i+1} = \lambda E_i$$

where λ was calculated (Section 3.1) to be

$$\lambda = 1.02717$$

Therefore the spacing between the eccentricities E_{i-1} , E_i and E_{i+1} can be calculated as follows:

$$E_i - E_{i-1} = (1 - 1/\lambda) E_i = 2.65 \times 10^{-2} E_i$$

$$E_{i+1} - E_i = (\lambda - 1) E_i = 2.72 \times 10^{-2} E_i$$

Hence the difference in spacing is

$$7 \times 10^{-4} E_i$$

The maximum eccentricity is less than half the dimension of the image which in this case is 256/2. Therefore the maximum difference in spacing is

$$7 \times 10^{-4} \times 256/2 = 0.09$$

of the sampling distance on the original image. Comparing this value with the absolute value of the spacings, then the spacing of the consecutive eccentricities can be considered to be locally constant. The radii of the processing areas at the above eccentricities are R_{i-1} , R_i and R_{i+1} respectively. The relation between the radii of the processing areas at the consecutive eccentricities is:

$$R_{i+1} = \lambda R_i$$

Using the same criterion as for the spacing of the eccentricities, the difference between the radii of the neighbouring processing areas is given by

$$7 \times 10^{-4} R_i$$

The largest size processing area in the peripheral region had a radius of 4 pixels. Therefore the maximum difference between the radii of the neighbouring processing areas is

$$7 \times 10^{-4} \times 4 = 0.003$$

of the sampling distance on the original image. Comparing this value with the radii of the processing areas, then the radii of the neighbouring processing areas can also be considered to be locally constant.

In super-imposing the processing grid on the rectangular sampling grid of the images, the centers of some of the processing areas do not exactly map onto an existing sample point of the image. Hence the centers of these processing areas have to be moved to the nearest available sample point on the image. This movement is referred to as the *quantization* of the co-ordinates. Quantization of the co-ordinates can introduce some error at the filtering stage of the proposed model, as explained below.

Filtering

It was explained in Section 2.3.1 that the DOG filter cannot provide any information about the true position of the edges that are closer than $0.97W$, where W is the diameter of the central lobe of the DOG function. The distance between the centers of the neighbouring processing areas in the peripheral region of the proposed model was evaluated to be $\sqrt{3}W/2$ ($\approx 0.87W$) which is less than the cut-off frequency of the filter (Section 2.3.1). Therefore no under-sampling occurs. The maximum quantization error occurs when both of two neighbouring processing areas (processors 1 and 2 in Figure 15) have to be moved in opposite directions, along the line connecting their centers to the sample points that are 0.7 units[†] from the calculated position of the centers of the processing areas. For this particular case, the quantization is 0.7 units for each processing area,

[†] the unit of measurement is the sampling distance on the image

Figure 15, Direction of the movements of the centers of the processing areas 1 and 2 due to the quantization of the coordinates.

which results in a total of 1.4 unit. The smallest processing area has a diameter of $W=4$ pixels. Therefore the quantization error E , is given by

$$E = 1.4, \text{ or}$$

$$E = 0.35 W \text{ (for the worst case)}$$

which results in a distance of

$$0.35W + 0.87W = 1.12W$$

between the centers of the neighbouring processing areas. If two edges of spacing $0.97W$ (which is the detection limit of the processors), both lie some where between the centers of these processing areas and perpendicular to the line connecting their centers [Figure 16], then neither of the edges will be detected by the processors 1 and 2. However, there is a good chance of detection of these two edges by any other six neighbours of the processors 1 and 2. But the above detection results in a smaller edge strength and a local orientation shift [Figure 17]. It should be noted that this type of error cannot occur repeatedly in a small region of the periphery. Also, it is note-worthy that this type of error is not of great importance since it can only happen in the peripheral region, and it has been repeatedly mentioned through out this report that the function of the peripheral region is to provide approximately information about the features in an image. Furthermore, the error of $0.35W$ is only for that region of the periphery with the smallest size processing areas. The worst case error is reduced to $0.17W$ for the processors at the far peripheral region. It should be noted that the spacing of the processing areas is about $0.87W$ ($\sqrt{3}R = \sqrt{3}W/2$) which provides a safety margin of $0.1W$ (10%) when compared with the cut-off frequency of the processors (Section 2.3.1).

Boundary following

Figure 16, Position of the centers of the processing areas after quantization. The lines between the dark and light regions determine the position of the edges.

Figure 17, The sign representation of the values of the sample points after the filtering process. Each sample point corresponds to the center of the associated processing area. The arrows show the direction of detected edges at point 1.

One other source of error is the method by which the boundary segments are grouped together (Section 3.4). It was mentioned that the segments with neighbouring end nodes were fused together and were considered to be part of the same boundary. This can cause error in differentiating the boundaries from the texture. One good example of this problem can be seen in Figure 12 (Section 3.4) where part of the texture on the lower right side of the hat has been mistakenly considered as part of the boundary of the hat. There are strong algorithms and methods by which these kind of problems can be overcome [Ramer-1975, Ballard and Sklansky-1976]. Only a simple algorithm was considered here to provide a means of comparison between the proposed model and the conventional vision systems.

4. SUMMARY AND CONCLUDING REMARKS

A new filtering technique, based on the physiological findings on the distribution of the receptive fields of ganglion cells, was introduced in which both the processes of filtering and sub-sampling of images were combined into one single step. A mathematical representation of the combined process of filtering and sub-sampling was developed for computer implementation. The technique resulted in a significant reduction in the number of multiplications involved in the band-pass filtering of images with a DOG (Difference of two Gaussian) function. This reduction was shown to be proportional to the square of the ratio of the sampling frequency of the image and the high frequency cut-off of the filter.

The new filtering technique was utilized in the development of a retinal image processing system in which the resolution with which the images were processed increased towards the central area of the image. The design procedure of the processing grid was such that the extent of the foveal region and the smoothness of the resolution change in the peripheral region were independent parameters. Therefore one could be changed without affecting the other.

The impulse response of the filters was chosen to be a DOG function. This function has a circular-symmetric spatial profile with a band-pass characteristic in the frequency domain. The DOG function resulted in the suppression of the uniform intensity profiles and the enhancement of the intensity changes in the images. The circular-symmetry property of the DOG function was also utilized in the development of algorithms to reduce the number of multiplications involved in the filtering of images. This reduction in the number of multiplications was

shown to be proportional to the radius of the DOG function.

The processing grid was based on a hexagonal distribution of the processing areas (filters). It was shown that each processing area was completely isolated by its six neighbouring processing areas.

The foveocentric property of the processing grid provided a three dimensional scale-space-map for analysis of lines (rather than two dimensional scale-space-map for analysis of points). This property was shown to be very useful in the development of algorithms to perform fast boundary following. Furthermore, this foveocentric property resulted in a significant reduction in the number of computations (multiplications, additions and comparisons) for boundary extraction and following. The overall reduction in the number of multiplications for extracting the boundary of the hat of the girl (in the example image) with the retinal image processing system compared with the conventional processing systems was shown to be

1. 39.3%, without utilizing the scale-space-map in the conventional image processing systems
2. 85.4%, with the utilization of the scale-space-map in the conventional image processing system.

4.1. EXTENSIONS TO THIS PROJECT

The foveal region of the processing grid was designed to have a processing area at each sample point of the image. This does not have to be the case. It was shown that the spacing of the processing areas can be increased to a few pixels

without introducing additional aliasing problems. Furthermore if the spacing of the processing areas in the foveal region is increased to $\sqrt{3}R$, where R represents the radii of the processing areas, then there would be no need to construct an intermediate region between the foveal and the peripheral region (namely the semi-peripheral region). Also by increasing the spacing of the processing areas in the foveal region, the number of multiplications required to perform filtering of images would be reduced by a significant amount, depending on the diameter of the foveal region. For the design parameters of Section 3.1, the number of multiplications in the foveal region would be reduced by a factor of 12 ($\approx 91\%$).

It was explained that the foveocentric property of the proposed model provides a three dimensional scale-space-map of the image. This property was a result of the assumption that, the radii of the processing areas increase linearly with eccentricity. This was formulated (Section 2.2.1) as:

$$R = K.E$$

If an extra assumption is made that, at each eccentricity E , all the processing areas of radii bigger than R are present, then the proposed model would also include the conventional two dimensional scale-space-map. This additional property of the model would become useful for stereopsis applications. It is actually suggested that the same criterion exists in the human eye.

Hardware

It is possible to design a hardware that can carry-out the first two stages of processing (Sections 3.1 and 3.2) in the proposed model. The coordinates of the processing grid can be calculated in an off-line fashion (eg; at the boot-up). These coordinates plus the coefficients of the associated DOG functions can be

stored in a look-up table. The absolute values of the eccentricities and the angular position around the origin of the processing grid are only calculated at the stage where the coordinates are first being generated. When storing these coordinates in the look-up table, the absolute values of eccentricities and angles are discarded. The storage of coordinates can be performed in an X-Y addressing system, where X represents the x^{th} eccentricity and Y represents the y^{th} processing area on that eccentricity. Hence the address [10,50] corresponds to all the coordinates and coefficients associated with the 50^{th} processing area at the 10^{th} eccentricity. The filtering of images can be performed in a pipe-line manner since each processing area is completely isolated by its six neighbouring processing areas. The frame-grabber would have to have a Dual-port RAM for pipelining the raw image data. With the above design considerations, processing time for the filtering of one whole frame of the image would be less than 0.1 seconds, using components such as the AMD29C101 μ -slice for addressing, the TI320C25-DSP for the arithmetic operations and the IDT7134 Dual-port RAM. Bearing in mind that the image is not completely filtered in the subsequent frames of the image.

REFERENCES

- H. Asada and M. Brady, "The curvature primal sketch," IEEE Trans. on Pattern Recognition and Machine Intelligence, vol. PAMI-8, No. 1, January 1986.
- J. Babaud, A. P. Witkin, M. Baudin and R.O. Duda, "Uniqueness of the Gaussian kernel for scale-space filtering," IEEE Trans. on Pattern Recognition and Machine Intelligence, vol. PAMI-8, No. 1, January 1986.
- D. H. Ballard and C. M. Brown. " Computer vision," Prentice Hall, 1982, Chapter 4, Sections 4.4, 4.5 and 4.6.
- D. H. Ballard and J. Sklansky, "A ladder-structured decision tree for recognizing tumors in chest radiographs," IEEE Trans. on Computers C-25, No. 5, May 1976.
- C. Braccini and G. Gambardella, "Linear shift-variant filtering for form-invariant processing of linearly scaled signals," Signal Processing 4 (1982), PP 209-213, North-Holland Publishing Company
- C. Braccini, G. Gambardella, G. Sandini and V. Tagliasco, "A model of the early stages of the human visual system: functional and topological transformations performed in the peripheral visual field," Biol. Cybernetics 44, 47-58 (1982).
- C. Braccini, G. Gambardella and G. Sandini, "A signal theory approach to the space and frequency variant filtering performed by the human visual system," Signal Processing 3 (1981) 231-240, North-Holland Publishing Company.
- S. B. Campana and D. F. Barbe, "Tradeoffs between aliasing and MTF," through personal communication, Naval Air Development Center Warminster, Pa. 18974.
- R. T. Chien and W. E. Snyder, "Hardward for visual image processing," IEEE Trans. on Circuits and Systems, vol. CAS-22, No. 6, June 1975.
- T. N. Cornsweet and J. I. Yellott, "Intensity-dependent spatial summation," Jorn.

- Opt. Soc. Am. A, vol. 2, No. 10, October 1985.
- J. L. Crowley and R. M. Stern, "Fast computation of the difference of low-pass transform," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-6, No. 2, March 1984.
- J. L. Crowley and A. C. Parker, "A representation for shape based on peaks and ridges in the difference of low-pass transform," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-6, No. 2, March 1984.
- B. Dreher and K. J. Sanderson, "Receptive field analysis," J. Physiol. (1973) 234, pp. 95-118.
- E. Dubois, "The sampling and reconstruction of time-varying imagery with application in video systems," in Proc. IEEE, vol. 73, No. 4, April 1985.
- R. Duda and P. E. Hurt, "Pattern classification and scene analysis," McGraw Hill, 1973, Chapter 7.
- O. D. Faugeras, "Fundamentals in computer vision," McGraw Hill, 1983, Chapter 2.
- R. E. Flory, "Image acquisition technology," in Proc. IEEE, vol. 73, No. 4, April 1985.
- N. Graham, J. G. Robson and J. Nachmias, "Grating summation in fovea and periphery," Vision Res. vol. 18, No. 18, pp. 815-825, 1978.
- J. J. Koenderink and A. J. van Doorn, "Visual detection of spatial contrast," Biol. Cybernetics 30, 157-167 (1978).
- M. D. Levine, "Vision in man and machine," McGraw Hill, 1985, Chapter 3.
- W. H. H. F. Lunscher and M. P. Beddoes, "Optimal edge detector design (I) and (II)," IEEE Trans. Pattern Analysis and Machine Intelligence, 1985.
- D. Marr, "Vision," Freeman, 1982, pp. 46.

- D. Marr and E. Hildreth, "Theory of edge detection," Proc. R. Soc. Lond. B 207, 187-217 (1980).
- R. M. Mersereau, "The processing of Hexagonally sampled two-dimensional signals," in Proc. IEEE, vol. 67, No. 6, June 1979.
- H. K. Nishihara, "Practical real-time imaging stereo matcher," Optical Engineering, vol. 23, No. 5, October 1984.
- L. O'Gorman and A. C. Sanderson, "A comparison of methods and computation for multi-resolution low- and band-pass transforms for image processing," personal communication, AT&T Bell Laboratories, Murray Hill, NJ, 07974.
- A. V. Oppenheim and R. W. Schaffer, "Digital signal processing," Prentice Hall, 1975, pp 239.
- L. Peichl and L. Wässle, "Size, scatter and coverage of ganglion cell receptive field centres in the cat retina," J. Physiol. (1979), 291. pp. 117-141.
- H. J. Reitboeck and J. Altmann, "A model for size and rotation-invariant pattern processing in the visual system," Biol. Cybern. 51, 113-121 (1984).
- G. Sandini and V. Tagliasco, "An Antropomorphic retina-like structure for scene analysis," Computer Graphics and Image Proc. 14, 365-372 (1980).
- R. H. Steinberg, M. Reid and P. L. Lacy, "The distribution of rods and cones in the retina of the cat," J. Comp. Neur., 148: 229-248.
- J. Stone, "A quantitative analysis of the distribution of ganglion cells in the cat's retina," J. Comp. Neur., 124: 337-352.
- U. Ramer, "Extraction of line structures from photographs of curved objects," Computer Graphics and Image Proc. 4, 81-103 (1975).
- J. M. White and S. G. Chamberlain, "A multiple-gate CCD-photodiode sensor element for imaging arrays," IEEE Jour. of Solid-State Circuits, vol. SC-13, No. 1 February 1978.

- D. R. Williams, "Seeing through the photoreceptor mosaic," through personal communication, Center For Visual Science, University of Rochester, Rochester, New York 14627.
- D. R. Williams, "Visibility of interference fringes near the resolution limit," J. Opt. Soc. Am. A, vol. 2, page 1087, July 1985.
- D. R. Williams, "Aliasing in human foveal vision," Vision Res. vol. 25, No. 2, pp. 195-205, 1985.
- A. P. Witkin, "Scale-space filtering," presented at the International Joint Conference on Artificial Intelligence, Karlsruhe, Federal Rep. of Germany, 1983.
- T. Y. Young and K. S. Fu, "Handbook of pattern recognition and image processing," McGraw Hill, 1986, Chapters 8 and 12.
- A. L. Yuille and T. Poggio, "Scaling theorems for zero crossings," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. PAMI-8, No. 1, January 1986.
- A. L. Yuille and T. Poggio, "Fingerprints theorem for zero crossings," J. Opt. Soc. Am. A, vol. 2, No. 5, May 1985.