CHAPTER I

INTRODUCTION

This thesis presents a practical microelectronic implementation of a visual segmentation scheme which consists of a network of coupled oscillatory cells. The implementation approach is inspired in neural activity observed in living beings. Based on previous work of biologists and computer scientists that focused on mimicking biological structures on computer simulations, a novel microelectronic approach of these schemes will be proposed. The aim is to achieve acceptable segmentation algorithms that need lesser power and area overhead thanks to low power microelectronic analog structures.

In this introductory chapter, we first overview this new branch of electronic engineering inspired in living beings known as neuromorphic engineering. Then, we show the complexity and we present problems of vision systems. After that, a rough overview of artificial vision segmentation schemes is presented. Since living beings seem to solve this complex problem apparently so easily, present knowledge of how they perform it is given and afterwards, some computing algorithms based on the same principles that animals are supposed to use. Finally we conclude this chapter with an overview of neuromorphic implementations applied to image processing and a brief outline and objectives of this thesis.

I.1 NEUROMORPHIC ENGINEERING

Since the dawn of technology, humans have dreamt of building machines that are able to solve the same problems that animals successfully do. With the advent of digital computers and artificial intelligence it seemed that we were able to reach this objective in few years. Nevertheless, experience has shown us that we are still far from this objective.

Modern computers can solve complex numeric problems a million times faster than human beings can do but they are still in their childhood when they have to perceive and understand a natural scene or sound. Even less evolved animals like flies or spiders are able to perceive their environment easily and fast enough to get food and avoid any danger. Since living beings evolved to such a good solution, an artificial approach that uses structures based on natural systems seems the way to follow.

Artificial Neural Networks [Lippman,87][Hush and Horne,93] have been the first approach to mimicking living being structures. These structures demonstrated their efficiency in solving classification and interpolation problems. Nevertheless further studies have shown that these networks have also severe limitations and the perception problem has not been solved yet.

Recent progress in neuroscience has brought us a clearer idea of the structure and organization of nerves and brain. This knowledge has led scientists to improve the model of the nervous system. Furthermore, advances in microelectronics have permitted the embedding of millions of transistors in a single integrated circuit. This progress led some years ago to the birth of a new engineering branch that mimics biological systems on silicon, it is known as Neuromorphic Engineering [Mead,89a].

Instead of using numerical and symbolic processing as digital computers do, neuromorphic systems exploit the physics of electrical circuits to perform operations similar to those found in the nervous system. Microelectronic analog circuitry allows the massively parallel, nonclocked and collective processing of these structures.

Compared to other approaches, neuromorphic engineering provides many advantages like:

- Speed: Parallel processing and transmission of data avoids an important bottleneck in serial processes.
- Size: Implementing an algorithm in a single chip reduces considerably the area overhead that is needed in other approaches as a microprocessor implementation. Moreover, various modules can be implemented in a single chip increasing system integration.
- Power: The possibility of using analog circuitry, which can operate in subthreshold region, instead of high-speed digital gates, reduces considerably power needs.
- Analog nature: Real world data is analog and no A/D conversion is needed, reducing the complexity of the circuitry.

However neuromorphic engineering faces some drawbacks.

- Full custom design: Neuromorphic circuits implement specific algorithms in a limited area, thus, off-the-shelf circuits cannot be used. This involves full custom design processes and they are slow, error-prone and more expensive than high level design techniques or programming a standard microprocessor.
- Use of a non-conventional technology: At present, efforts to improve microelectronic technology are focused on digital circuitry that covers the great majority of the market. The use of analog circuitry has some problems related to the use of a technology that is not well characterized. However, analog design techniques are improving to accommodate analog circuits in digital designs using cheaper and more compact digital processes.
- Low precision: Precise analog circuitry consumes a lot of area compared to its digital counterpart. Analog technology is only advantageous when high precision is not required or it can be compensated by design techniques. Furthermore, design techniques should account for this lack of precision and use robust algorithms.
- Lack of flexibility: To reduce area overhead, circuits are hardwired and cannot be programmed to change the algorithm for which they have been designed.
- Low Resolution: Full parallel implementations, although being very fast, require large areas to be implemented. As a result, present technology circuits cannot accommodate a large number of sensing and processing elements. Vision algorithms are especially sensitive to this problem due to CMOS photosensor large area overhead. This leads to low-resolution designs compared to conventional CCD cameras. However this is a problem related to technology maturity and it will probably be solved in the next years.

For these reasons, digital computers are preferable for developing and testing new algorithms while analog neuromorphic implementations are more suited for implementing these well-tested and error-free algorithms to portable systems.

I.2 THE VISION PROBLEM

Vision systems must deal with the problem of interfering geometrical and physical properties of surfaces under analysis. The available data consists of two-dimensional arrays of light intensity captured by sensing devices, which we call the image or a temporal sequence of some of them. When physical properties as distance, orientation, texture or motion are understood by the vision system, it can react properly as to navigate through the environment, manipulate different objects, recognize their surroundings or even reason about them.

However, these problems are usually very complex and involve a great amount of data processing. In addition, they are considered to be ill posed problems because they may admit from no solution to infinitely many solutions. To cope with this complexity, the whole task is split in different stages. As a first approach to this, it can be said that first, the image must be detected by photosensing devices (as retinal cells in living beings or CCD and CMOS sensing devices in artificial systems) and filtered in the spatial and temporal domains to reduce noise and extract some basic characteristics. Then, this preprocessed information must be segmented in coherent objects and organized in hierarchical structures to be understood in higher stages. In addition, as so much information is stored in any image, in most cases, some attentional layer must concentrate the attention of the system to a specific part of the scene while ignoring the rest. For these reasons, it is easy to admit that understanding a natural image is a very complex process that requires different levels of processing and it is very expensive for artificial systems in terms of computer load.

First levels of visual processing, also known as early vision, consist in acquiring the scene, filtering image noise and extracting some characteristic features of the image as edges or motion. Unlike higher level vision processes, first stages of early vision are mainly a bottom up approach, that is to say, they only rely in optical data captured by sensors and not in a previous knowledge of the scene to be analyzed. Some examples of it are noise filtering or motion detection. On the other hand, higher levels of vision use top down approaches. They need some knowledge of what they are 'seeing' as to recognize objects or distinguish dangerous obstacles in a navigation system. Other stages, which can be regarded as higher stages of early vision or lower stages in high level vision as segmentation and attention, may use a bottom up or a top down approach. Bottom up approaches are simpler because they do not need higher levels of visual processing to be implemented while their results may not be accurate. Top down approaches may obtain better results but they are more complex because they need some knowledge of what it is being seen and available data in these stages is still too large and redundant. For these reasons, in multiple visual schemes, these intermediate levels use both approaches. First, a bottom up approach to infer some results. Then, based on previous knowledge of what can be seen, these results must be fitted or parameters in the bottom up approach adjusted.

In living beings, early vision tasks are solved in the retina. Rods and cones are the two types of photosensitive cells but there are other cells in this organ. In addition to detecting light, the retina makes a preprocessing of the information through other cells as bipolar and ganglionary cells. Then, information is lead through the optical nerve to the thalamus and the visual cortex where higher processing vision stages take place.

Traditional artificial approaches usually detect images with a dedicated system as a CCD camera. Then, this information is digitized and led to a processing unit as a DSP, a microprocessor or a traditional neural network to be filtered, extract some characteristics and/or make other signal processing until the task for which it has been designed is accomplished. One of the main drawbacks of this approach is the need of a high speed link between detecting and processing elements, besides being in most cases serial processing schemes, which make them slow for high resolution images.

In addition to traditional artificial approaches, neuromorphic engineering presents new possibilities with its advantages and particular problems, which are presented in this chapter. For early visual tasks that take place in the retina as photosensing, filtering and characteristic extraction, they have been extensively studied in the literature and a large number of neuromorphic implementations exist. Few examples of this effort are: [Delbrück,93], [Harris et al, 90], [Koch and Li,95] [Mead,89b] among others. However, neuromorphic implementations of intermediate levels of processing that take place in brain areas as attention and scene segmentation have not been studied under this perspective as much as retinal stages. Since these intermediate levels are also necessary for a complete image processing and understanding system, if they are embedded in the early vision system, transmitted data could be reduced and also computer load in higher stages. This may lead to a substantial reduction of system complexity and suit compact and low power system needs. This thesis focuses on the electronic implementation of such intermediate algorithms in view of embedding them in other circuits to perform complete vision tasks.

I.3 COMMON SCENE SEGMENTATION TECHNIQUES

Scene segmentation is one of the key stages in visual perception processes. The aim is to divide an image in different parts that correspond to physical objects in the scene. Then, results can be parsed to higher levels of processing for recognition, classification or interpretation.

Achieving a good segmentation is not a trivial problem since there are different kinds of images. A non-exhaustive classification is given below [Pal and Pal,93]:

- Light intensity images (LI): The light intensity (gray level or color) is represented in each pixel of the scene. These are the most common types of images. The ones we see in our daily experience.
- Range image (RI): Where each pixel represents the depth (distance from the viewer) of different points of the scene. These images are used in radar applications or stereoscopic vision.
- Nuclear Magnetic Resonance Image (MRI): These images represent the intensity variation of radio waves generated by biological systems when exposed to radio frequency pulses. They are mainly used in medicine applications.
- Thermal image (TI): Each pixel represents the temperature of each point of the object. They are obtained with IR sensors.

Hundreds of segmentation techniques have been published in the literature, but no single method can be considered good for all images. In addition to this, each method is not equally good for all kinds of image, i.e. an algorithm developed for MRI images, could not properly be used to another class of image as an ordinary LI. Segmentation algorithms are basically ad-hoc and rely on some particular properties that are not common to all images. They differ in the way they emphasize one or more desired properties and in the way they balance one property against another. Therefore, the choice of the optimal segmentation algorithm depends on the computer capacity available and the kind of images to segment.

Obviously, this step is integrated to a whole recognition system. It means that various segmentation schemes can be used simultaneously, at expense of higher computational load. Then results of these schemes can be compared at higher processing levels and extract the best of them. Also higher levels of processing can feedback lower levels to improve segmentation.

A complete survey of image segmentation techniques is beyond the aim of this work but a quick overview is given below.

Various classifications exist in the literature [Pal and Pal,93] [Nevatia,86] [Haralick and Shapiro,85] [Fu and Mui,81]. Mainly, scene segmentation algorithms are Edge or Region Based whether they detect sharp changes in the image or they group pixels of similar characteristics, respectively.

I.3.1. Edge Detection Schemes

Edge detection techniques are based on the detection of a discontinuity. Boundaries are placed where there is an abrupt change in a certain feature. Algorithms can be sequential (the decision of an edge pixel is dependent on results obtained at previously examined pixels) or parallel (the edge detection operator can be applied simultaneously everywhere in the scene). Usually, detected edges are not continuous or they are placed on a discontinuity of the image that does not correspond to an object boundary. Therefore, a following step is connecting these edges together to form closed curves and deleting edges that not belong to any boundary. This last step has a sequential nature.

I.3.1.1. Gradient Operators

The simplest method is to compute the gradient of the image intensity. Then, edges are said to be present when the magnitude of the gradient exceeds a certain threshold. Gradient can be approximated by differences in any two orthogonal directions of the image or using a more complex operator as Roberts' [Roberts,65]. They require low computation cost but only perform reasonably well in images of low noise and texture. Operators that are more complex exist in the literature. They have a better performance at the cost of a higher computational load.

I.3.1.2. High-Emphasis Spatial Frequency Filtering

Sharp changes in image intensity are associated with higher spatial frequencies. Thus, high-pass filtering the image enhances these changes and edges can be detected because they are usually associated with these frequencies. Filtering is achieved by applying the Fourier transform to the image, applying the filter in frequency domain and taking the inverse transform of the result. The main problem of this method is the filter design.

I.3.1.3. Functional approximations

This method considers edge detection as an approximation problem. Small parts of the image are considered as noisy analytical functions and these ideal functions are used to compute the derivative and obtain the edges.

I.3.1.4. Template matching

Template matching is not only used in segmentation but also in other areas as object extraction. In this approach, some ideal templates are convolved with the surroundings of each pixel of the image. These templates represent ideal boundaries in different orientations and high output is expected where the image is like the mask. Note that the output is high at edge-pixels and at their neighbors. This property can be used to detect edges with more confidence and must be taken into account to pick only maxima to avoid thick edges. Mask size is an important choice. The bigger the mask is the better performance on low contrast edges and against noise but lower precision and higher computation cost.

An important drawback of this method is that it fails when there is a great variation of patterns to be matched.

I.3.1.5. Second derivative methods

First derivative methods (as gradient schemes) respond erratically on a ramp intensity profile. Second derivative algorithms eliminate this difficulty. For a step edge, the second derivative is zero at the step and has a positive and a negative value in either side. The problem reduces to locating zero crossings.

The best known second derivative method is the Laplacian operator [Eq. I.1].

$$\nabla^2 = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \qquad \text{Eq. I.1}$$

This consists of the sum of the second derivatives, thus providing information on the "acceleration" in the variation of the light intensity of the image. The main drawback of this method is its response to noisy pictures. Noise produces higher Laplacian values than edges.

To improve this method, the Laplacian Gaussian (LG) operator has been introduced. It is the result of applying the Laplacian of Eq. I.1 to the two-dimensional Gaussian distribution with standard deviation σ:

$$G = e^{-(x^2+y^2)/(2\pi\sigma^2)} \qquad \text{Eq. I.2}$$

The Gaussian part of the LG operator blurs the image wiping out all structures smaller than σ. A combination of four LG operators with different deviation has been suggested because it seems to be a good model of human visual processing. Then, the four results can be combined at higher processing layers.

On the other hand, the LG operator is computationally expensive and it is approximated by a difference of Gaussian functions (DOG):

$$DOG = e^{-(x^2 / 2\pi\sigma_e^2)} + e^{-(y^2 / 2\pi\sigma_i^2)} \qquad \text{Eq. I.3}$$

I.3.1.6. Sequential Techniques

When using sequential techniques for edge detection, the result at a point depends on results at previously examined pixels. For a good sequential procedure, one must be careful in selecting the initial point, the criterion to determine when the algorithm must finish and how the results obtained may affect the selection of the next point to process and the way it is affected.

Some sequential techniques are heuristic search, dynamic programming and guided search.

I.3.2. Region Segmentation Schemes

In Region segmentation schemes, the scene is segmented based on connected set of pixels that share a common property as color or intensity.

I.3.2.1. Thresholding

This is the simplest method of region segmentation. All pixels that share a common property, as intensity, are grouped together. These groups give us the desired scene segmentation.

This method is suited for images containing homogeneous objects against a high-contrast uniform background.

The main problem of this method is to find the proper threshold and various techniques have been proposed. Basically, they can be classified in global, local and dynamical threshold [Fu and Mui,81]

Global threshold: The same threshold applies to the whole image. It can be computed from selected typical images or from statistics of the analyzed image. One of the simplest is computing the histogram of intensity values (x-axis measures the gray-level and y-axis measures the number of pixels with that gray-level). Then histogram valleys are detected and their values are selected to become thresholds. This method can be improved by processing the histogram at expense of higher computational load.

Local threshold: Threshold varies throughout the image and it depends on the characteristics of the neighborhood of each pixel.

Dynamical threshold: Threshold varies for each pixel or group of pixels and depends on the characteristics of the neighborhood and the position of the pixel in the image.

I.3.2.2. Clustering

Clustering is the multidimensional extension of thresholding. Two or more features are used, as gray-level, color, texture of a local neighborhood, the result of applying a template operator or any other characteristic that can be helpful. Then points in the feature space are grouped into clusters and mapped back to the original spatial domain to obtain the segmented image.

Grouping of points in a multidimensional space is not as easy as in the univariate histogram clustering of thresholding and specific techniques for it have been developed.

I.3.2.3. Region growing

The objective is to merge regions of the image (or even single pixels) in function of some of their characteristics. Regions can be merged using different criteria:

Single Linkage Region Growing: Each pixel is considered a node of a graph and neighboring pixels are joined with an arc if they are similar enough. The result is maximal sets of pixels that all belong to the same connected component. The similarity criterion is very important. The simplest is the pixel difference, but others have been proposed.

This method is the simplest and needs images with well-defined objects and little noise. The main problem is that only one arc is needed to join neighboring clusters of the image so unwanted merging is very common. However, boundaries are placed in a spatially accurate way.

Hybrid Linkage Region Growing: This method is more powerful that the single linkage. Each pixel is assigned a property vector that depends on the NxN neighborhood of the pixel and the similarity between pixels is computed through this vector. Therefore, pixels are similar because their neighborhoods are similar. This vector can be computed through applying edge detector operators to the image or comparing textures.

This method is more robust to noisy data and can segment more complex images but it is computationally more expensive.

Centroid Linkage Region Growing: This method does not compare similarity between pixels but pixel characteristic is compared to existing neighboring segments, which are not complete yet. If pixel characteristic is similar enough to the mean characteristic of any of existing objects, this pixel is merged to that object. If it is different enough, a new object is created. Scanning of the image is sequential and predetermined and it can lead to different segmentation solutions.

Single pass algorithms have problems with some object shapes and may segment them in two objects. A V-shape segment is a good example. Solutions to this problem can be double pass algorithms or a combination of single-pass algorithm followed by a connected component algorithm.

Hybrid Linkage combination: This technique extracts the best of linkage methods detailed above. Centroid linkage computes pixels that are not on edges and single or hybrid linkage is used to compute edges. It avoids excessive merging due to single linkage and boundaries are spatially accurate.

I.3.2.4. Spatial Clustering

Haralick and Shapiro [Haralick and Shapiro,85] call spatial clustering a technique that mixes clustering and region growing. Region growing is computed considering histogram calculation.

I.3.2.5. Split and Merge

This method consists in dividing zones of the image that are not homogeneous enough and merging zones that are similar. Homogeneity and similarity are computed through a proper feature as the difference of the largest and the smallest gray tone intensity.

I.3.3. Color images

Color images are a special case of multispectral scenes and algorithms for this kind of images usually can be used. Color can be described by the distribution of the three-color components (RGB) or psychological components (hue, saturation and intensity) but combinations of these sets can also be used and found more effective as the Karhunen-Loave transform.

However, experience indicates that information in color components is highly correlated, thus, extra computation for color edge detection is not cost-effective in general cases.

I.4 BIOLOGICAL PRINCIPLES OF OSCILLATORY NEURAL NETWORKS

Although the first approach to neural networks simplified neurons to simple comparator units [Lippman,87], more accurate models demonstrated that oscillations and pulsed behavior of neurons are also an important computing characteristic, e.g. [Maass and Bishop,99]. For this reason, the scientific community is doing a great effort to understand them through microelectrode recordings and simulations.

Vision is one important field of study for oscillatory neural networks. It is well known that human visual system captures light from its surroundings through photosensory cells of the retina. At this level, the representation of the exterior is just a set of pixels of different colors and intensities. However, Gestalt theory [Rock and Palmer,90] states that the external world is not perceived as pixels in our minds. We 'see' differentiated objects that follow perception laws (Proximity, Similarity, Good Continuation, Connectedness, Common Fate, and Prior Knowledge). Hence, there exists a segmentation and binding process of different properties of the image seen in higher levels of mind.

When an artificial system concentrates its attention on something, it must follow the same process except that the external information are pixels sensed by a CCD or CMOS camera or a digitized photograph instead of being nervous pulses generated by light sensitive cells from the retina. Afterwards, this information is used in a higher level stage to recognize or follow the objects.

When light emitted from one object impinges retinal cells, these cells activate neurons of different areas of visual cortex that correspond to a distributed representation of its properties (color, shape, motion, etc.) [Engel et al,91a][Eckhorn,97][Eckhorn et al,88][Gray et al,89]. It seems that visual cortex neurons are specialized in different visual characteristics and they are grouped in different areas [Livingstone and Huble,88]. It is still unknown how these features are grouped to perceive the object as a whole, but it has been suggested that this association is made by an oscillatory mechanism of the electrical activity of neurons.

Experiments on animals (especially cats and monkeys) demonstrate the existence of correlation in electrical activity of neurons in the visual cortex when a simple stimulus is applied to the retina. These correlations suggest that image segmentation is carried out in these zones by means of a synchronization mechanism of neuronal activity.

Models of neuron behavior have been simulated and they proved that correlation of electrical responses depends on the spatial coherence of stimulus. These networks have been simulated using different neuron models: binary neurons with stochastic behavior [Kappen,97] or analog and deterministic neurons [Sporns et al.,89].

Engel et al. [Engel et al,91a] have demonstrated experimentally the existence and characteristics of these oscillations. When an animal retina is stimulated with the image of a bar, the activity of a group of neurons in visual cortex synchronizes. But, if two bars with different orientation appear in the image, two groups of neurons emerge. Electrical activity is correlated within each group and uncorrelated between groups. Hence, each group of neurons is sensitive to a certain orientation.

Frequencies of these oscillations range from 40 to 60Hz [Gray et al,89] [Engel et al,91a] or from 35 to 80Hz [Eckhorn et al,88]. Eckhorn and his team announced in the same paper that coherent signals exist within a vertical column, between neighbor hypercolumns and between different areas of cortex. It is reasonable to assume that synchronizing connections exist between neurons that are sensitive to the same characteristic, whether they are in the same cortex area or not. In addition to this, this correlation is also found between both cerebral hemispheres. Experiments show that synchronization is achieved through connections that cross *corpus callosum* because synchronization disappears when this part of the brain is cut [Engel et al,91b].

Other scientists suggest that information is based on chaotic behavior of oscillatory neurons [Freeman,92] [Skarda and Freeman,87]. Their work is based on electroencephalographic (EEG) potentials; that is to say, the activity of groups of neurons is measured instead of individual potentials. From this point of view, they have concluded that information processing of olfactory system is carried out using the envelope of electrical activity of groups of neurons. Information lays in the kind of chaotic behavior that neurons generate instead of being the exact form of signals.

Based on that premises, the olfactory bulb has been modeled [Li and Hopfield,89] [Freeman,87] using non-linear oscillator networks that generate chaotic behavior.

Chaotic oscillatory networks have been used for pattern recognition [Yao and Freeman,90] [Shimoide and Freeman,92]. In the first paper, a mathematical model is presented. It has interesting properties as scalability, spatial coherence and it is independent of initial conditions. The second paper shows an application example of Japanese character recognition. The system can recognize five Japanese vowels if the acoustic input signal is previously preprocessed.

Also motion can be a characteristic to segment objects of an image from the background. In [Reichard et al,83] the domestic fly visual system is analyzed and a model of it is proposed. Finally, results from simulations are compared to measures of biological systems.

I.5. OSCILLATORY SEGMENTATION SCHEMES

I.5.1. Introduction

Making the most of findings presented in the previous section, computer scientists have developed various algorithms for visual information processing based on arrays of oscillators that are supposed to use the same principles as living being structures do. Some of them are analyzed in this section.

Mainly, in these schemes, each oscillator of the network is associated with a pixel of the input image or a feature of it. The objective is that pixels belonging to the same object oscillate synchronously, pixels that belong to different objects oscillate out of phase and pixels that do not belong to any object do not oscillate. At present, different research groups have focused their activities to that subject and a brief review is given.

Baldi and Meir [Baldi and Meir,90] studied the possibility of using coupled oscillators for texture discrimination. Oscillators under study were linear and they were applied to segment synthetic images of textures. Input was preprocessed through Gabor filters before exciting oscillatory network. This paper stated some of the most important features of this new paradigm of computing:

i. It is robust and fast because few oscillations are needed

ii. Hardwired implementations seem very straight

iii. However, little information is kept in oscillations

iv. Additional machinery is needed to process and route information

I.5.2. Malsburg's model

In 1986, von-der-Malsburg and Schneider [Malsburg and Schneider,86] proposed a system to solve the Cocktail Party Problem. This is, the problem to differentiate voices in a noisy and interfering environment, as it can be found in a cocktail party where many people are talking at the same time, music is playing in the background and other noises are also present. Even in these adverse conditions, anyone can understand his interlocutor. Malsburg and Schneider system is a one-dimensional matrix of excitatory cells plus an inhibitory one also connected to all other cells. The network receives two stimuli with 10 spectral components each, hence, there are 20 inputs to the network. It should be stressed that this network includes a global inhibitor to produce oscillations but also to achieve desynchronization and maintain it, a very important element in posterior models. However, this network uses global connectivity, that is to say, each cell is connected to all other cells, which means a non practical number of connections in bigger networks. Connections are on the order of n^2 where n is the number of cells. The aim was of this paper was to demonstrate that synchronization/desynchronization could be achieved and the network was only applied to synthetic images.

A subsequent paper of the same group [Malsburg and Buhmann,92], defines a model made of individual oscillators, partial inhibitors and a global inhibitor. The network is a three-dimensional matrix of coupled oscillators that are sensitive to localized input from the sensory field and encode local feature types. The network is made of layers of two-dimensional arrays and a specific feature of the input excites each layer. Connections are global within a layer and several inhibitors are also used. This means that there exist a large number of connections, which makes it difficult to implement. They show the possibility of synchronizing/desynchronizing oscillators although their network is not applied to real images. Applications to real world data can be found in [Vorbrüggen and Malsburg,95]. Features of gray level images are used as input to the network and objects are successfully segmented in few cycles. However, manual intervention is required to code input data.

I.5.3. LEGION

Using the same principles as in networks introduced above, Wang and Terman proposed the Locally Excitatory Globally Inhibitory Oscillator Network (LEGION) [Wang and Terman,95]. The most important advantage of this network from the point of view of its implementation is that it has few global connections and all other synapses are local. This reduces considerably the difficulty of implementing it.

The network is composed of a 2-D array of coupled nonlinear oscillators, each one being described by means of a pair of ordinary differential equations. Inputs to these cells are input stimuli, local excitatory connections, noise and a global inhibitory connection. Input stimuli establish excitatory connections when oscillators are associated with pixels that belong to the same object. When these couplings are established, they are responsible of synchronizing oscillators while a global inhibitor is responsible of desynchronizing them. When two oscillators are excitatory coupled, excitation is stronger than inhibition and oscillators synchronize. However, when there is no excitatory coupling, inhibition desynchronizes oscillators. After few cycles of operation, the network reaches an oscillatory stable state that stores enough information to distinguish objects.

In addition to the 2-D oscillator array, there is an extra cell, the global inhibitor, that is coupled to all cells of the network. This global cell shows an important improvement in network segmentation skills in front of other schemes without global connections. The use of the global inhibitor is a good trade-off between the cost of the global connectivity and performance.

It is important to stress the importance of noise in this network. Since all simulated oscillators are equal, an unstable equilibrium state may be reached. Then, noise is responsible of breaking this equilibrium producing small differences between cells. These differences allow the global inhibitor to desynchronize oscillators that are not associated with the same object. Further details on LEGION scheme are given in chapter II.

Various kinds of differential equations have been used to describe basic oscillators. A hyperbolic tangent and a cubic are used in most cases [Wang and Terman,95] [Wang and Terman,97] [Wang,96a] [Shareef et al.,99] but also sigmoids [Wang and Buhmann,90], linear oscillators [Wang,95], integrate-and-fire oscillators [Campbell and Wang,98a] and Wilson-Cowan systems [Campbell and Wang,96]. This shows how the choice of oscillators is not essential [Shareef et al.,99]. This characteristic is very important in an electronic implementation because oscillators can be implemented with a higher degree of freedom.

Most papers simulate LEGION on synthetic binary images, especially the first ones, but gray level real images have also been used [Wang and Terman,97] [Shareef et al.,99]. Models for binary images stimulated oscillators associated with foreground pixels and relaxed background oscillators. On the other hand, for gray level images, all oscillators are stimulated and synapses are proportional to gray level correlation of neighbor pixels. Hence, all oscillators oscillate. Up to our knowledge, LEGION has been applied to light intensity images [Wang and Terman,97], MRI [Shareef et al.,99] and range images [Liu and Wang,97].

An important characteristic of the last papers mentioned above is that they include a new term in oscillator equations. This is the lateral potential. This term excites oscillators that belong to big objects. Thus, in real noisy images segmentation is also successfully achieved provided that noise is low.

To speed up simulations considerably, a new algorithm (G-LEGION) has been developed. It uses a higher level model of oscillators that ignores their precise dynamics. Although this simplification is accurate enough to test dynamics of an ideal system and obtain successful segmentation results, it is too rough to obtain the exact waveform of oscillators and test its non-idealities as time delays.

Finally, LEGION has been also applied to attention problems [Wang,96b]. This system is able to concentrate its attention to the biggest object of an image.

A major drawback of LEGION is the high computational load when it is simulated on a computer. Each oscillator is a differential equation that should be solved. Their behavior is very similar to an astable multivibrator. Dynamics are very fast when changing its output state and very slow the rest of the time. A 32x32 network has 1024 basic oscillators plus an inhibitor, thus, 1025 equation systems must be solved for a small image. On the other hand, oscillators are easily implemented in microelectronics and no differential equation has to be solved because circuit laws perform it. Thus, in spite of LEGION being difficult to implement on a computer for real time applications, it seems to be a good candidate for a VLSI implementation.

In physical implementations, delays of oscillators and synapses play also an important role. First models of LEGION did not consider delays because they were not implemented in the algorithm. However they must be considered when using real oscillators to implement the network. Campbell and Wang [Campbell and Wang,98b] simulated a network with time delays, and demonstrated that delays degrade synchrony and they can even prevent synchrony if they are large enough. Chen and Wang [Chen and Wang,97] [Chen and Wang,98] also studied the problem and gave a biological interpretation of it.

Brown and Wang used LEGION to segregate spoken sounds [Brown and Wang,97] [Wang and Brown,98]. These models filtered the input sound using a cochlear model, then autocorrelated filtered sounds and obtained a three-dimensional representation of them along time, frequency and lag. Then, segmentation and binding using LEGION is carried out to segregate sounds or speech from interfering noise.

Simulations shown in papers demonstrate good performance of models. However, preprocessing of signals is computationally expensive and the three dimensional representation of sound forces a serial processing on a two-dimensional representation that makes its implementation more difficult.

I.5.4. Other Oscillatory models

Zhaoping Li introduced another oscillatory model of the primary visual cortex [Li,98a] [Li,98b] [Li,97] for contour enhancement and edge binding. Her network is made of oscillators whose dynamics are defined by a non-linear system equation. Inputs are edges of an image. When edges are close to each other and they have similar orientation, neighbor cells associated with these edges are excited and their oscillation grows. Thus, oscillators associated with straight line have a big amplitude oscillation, larger than oscillators associated with close contours, while oscillators associated with isolated edges stay silent. Simulations on synthetic images show good results and this model works reasonably well on gray level real images.

Labbi et al [Labbi et al,97] used a network of oscillators embedded in a three-layer architecture to detect salient regions of an image. There is a layer for the input data, another layer for the feature map and finally an attention map, which feeds back the feature map. Basic cells are FitzHugh-Nagumo oscillators, which are derived from Hodgkin-Huxley neuron model. It has to be stressed that one important feature of these oscillators is that their frequency is almost constant and it is independent of input strength while their amplitude is strongly dependent of input. The aim of the network is to detect salient regions of the input image and simulations show good results when applied to real gray level images.

Bosch et al [Bosch et al,98] improved the aforementioned network. They used a four-layer scheme where a location map is added. Also spiking neurons are used instead of FitzHugh-Nagumo oscillators. Possibilities of spiking neurons for attention applications are studied in [Bosch et al,97].

I.5.5. Cellular Neural Networks

Although Cellular Neural Networks (CNN) are not oscillatory networks, they are included in this chapter because of their analog and cellular nature.

Since CNN's were first introduced by Chua and Yang [Chua and Yang'88a] [Chua and Yang'88b] in 1988, intensive research has been done in that direction.

A CNN is an analog dynamic processor array whose elements interact directly in a finite neighborhood. However, due to dynamic propagation, cells that are not directly connected interact with each other. Chua and Roska [Chua and Roska,93] defined a CNN as a n-dimensional array of mainly identical systems (cells) in which most interactions are local and all state variables are continuous valued. When a template is loaded and an external stimulus is applied to the network, it starts its evolution towards a stationary state, which is the result of the desired operation. Since CNN processes in parallel, a reduced time is needed to reach this state and thus, find the solution to the problem.

Microelectronic implementation of CNN's is straightforward due to their small number of connections, which is proportional to the number of elements that compose the net. Some CNN implementations are: [Cruz and Chua,91] 6x6 array focused on connected component analysis, [Dalla Betta et al,93] 10x10 simulated analogically programmable array, [Slot et al,96] simulated array for edge extraction and image half toning.

The main advantage of such networks is that they can be programmed to carry out a wide range of mathematical operations. A template specifies the interaction between each cell and its neighbors. However, this flexibility comes at the price of a very large basic pixel circuitry and thus its power needs.

Sample applications of such networks are: approximation of partial differential equations [Chua and Yang'88a], noise removal and edge detection [Chua and Yang'88b], connected component detection [Matsumoto et al, 90], etc.

I.6. NEUROMORPHIC VISION IMPLEMENTATIONS

Next, an overview of different early vision algorithm implementations using the neuromorphic approach is given. Typically, these implementations include photodetection and preprocessing stages in the same die. Obviously, as not all silicon area is dedicated to image acquisition and technology processes used are not specifically oriented to photodetection but to signal processing, performance of such circuits cannot be compared to dedicated image acquisition circuits, i.e. CCD’s. The aim of visual neuromorphic implementations is to optimize image processing capabilities, besides acquiring it, using as less power as possible and occupying the minimum silicon area while standard CCD sensors are focused on obtaining the maximum image quality. A typical example for it, it is Mead and Mahowald's silicon retina [Mead and Mahowald,88].

The reason of such an effort for this specific task is that vision is computationally very expensive. As standard hardware (i.e. computers) is inefficient in terms of area and power consumption, dedicated hardware is an interesting alternative when these constraints are important.

Although special-purpose digital hardware is a first approach to this problem, its power consumption and area needs are very important. Moreover, if a digital processor is not integrated in the sensing device, and important amount of data must be transferred from the camera to the central processing unit. This limits resolution and scan frequency.

On the other hand, if image quality is not the main objective, why not integrate acquisition and early image processing analog stages as the visual system does. It will free the central processing unit of doing computationally expensive tasks and also, it could reduce transferring speed. In addition to this, preprocessing simplifies the A/D conversion requirements.

First analog circuits for image processing were proposed in 1974 [Koch and Li,95] although technology was still not mature enough for practical implementations. In 1989, Mead [Mead,89a] published a book that stated the basics for this area.

Up to now, neuromorphic implementations have mainly focused on early visual tasks as image filtering, stereo vision and edge or motion detection. However, some efforts have been done in attention and segmentation.

I.6.1. Phototransduction

The first issue for an intelligent image sensing circuit is phototransduction. To cut down costs, most implementations use standard CMOS technology (it is the cheapest). CCD (Charge Coupled Device) technology is more expensive and it is reserved for applications that need high quality images. In addition to this, if analog circuits are placed near the CCD, sensor clocking may easily interfere in the normal operation of analog cells.

Another issue to take into account is that dynamic range of light in the natural world swings over eight orders of magnitude, thus, a phototransducer that exhibits logarithmic output is desired. A simple photodiode can logarithmically compress current (induced by light) into a voltage. However, its response is very slow for low intensities and device mismatch can be so important that errors become larger than signal.

The solution to this problem is adaptive phototransduction. Delbrück and Mead [Delbrück and Mead,94] designed a phototransducer that can adapt to slow changes of six orders of magnitude in light intensity within the image, while preserving high gain for transient changes in the image. Although it does not need complex clocking circuitry, an important drawback of this design is its area overhead. Delbrück's phototranducer is 55 times bigger than a simple CCD pixel. Thus, up to now, high resolution can not be achieved using this technology. However, it does not seem to be a serious problem. While human beings have one or two million sensing cells, simpler animals cope with less photoreceptors. For instance, a common house fly has less than 7000 picture elements and can perfectly survive in hostile environments. This demonstrates that for many visual tasks, high resolution is not needed.

I.6.2. Image Filtering

Since filtering a two-dimensional signal as an image is very expensive in terms of computing cost, parallel architectures have been developed. Most of these architectures convolve the image with a Laplacian operator (a common operation in visual processing to extract important features of an image) and only use local connections. As this reduces connections considerably, electronic implementations are feasible.

Greatly simplifying the structure of a real image, when two neighboring locations are very similar, it is assumed that there is no information because they belong to the same object. Therefore, any difference between these elements is due to noise and it should be reduced. However, when difference between two neighboring pixels is high, they must belong to different objects and differences should be stressed. If the voltage across two pixels is small, their connection must behave as a conductance proportional to their gradient. However, if the voltage is high enough, the connecting element must saturate as a Horizontal Resistor [Mead,89a] or drop current to zero as a resistive fuse [Harris et al,90]. This mechanism performs a local gain control in contrast to global gain control of standard CCD cameras. This allows seeing details on shadowed parts of an image and also detecting edges.

One of the first designs is Mead's and Mahowald's silicon retina [Mead and Mahowald,88] that presents a resistive hexagonal grid that can compute a discrete approximation of the Laplacian operator. Since resistive elements are difficult to implement on CMOS technology, transistors are used instead. This design allows controlling resistance by an external input and its devices saturate at high voltages, which improves circuit behavior. This design uses a parasitic vertical bipolar transistor as photosensing device, which is a by-product of the CMOS process, and feeds its current to a circuit with an exponential current-voltage characteristic. Thus, voltage out of the photoreceptor is over four orders of magnitude of incoming light intensity. In addition to this, Mead's retina also performs a temporal filtering of the image, removing temporal low frequency information.

An interesting design for image segmentation has been presented in [Bair and Koch,91]. It takes the difference of two resistive-networks that smoothes the input of photoreceptors and finds zero-crossings. This network performs a difference of exponentials, which is an approximation of the difference of Gaussians. Simulations show its good performance to detect edges on a two-dimensional image. A one-dimensional version of the algorithm has also been implemented. Simulations of a design that also stores edges in a digital memory were presented in [Yi et al,97].

An important amount of work has been done in silicon retinae and designs have been considerably improved to reduce mismatch [Mead'89b] based on floating gate technology, to improve the approximation of the Gaussian response with negative impedance converters [Kobayashi et al,95], the foveated chips (chips with increasing resolution in their center) [Wodnicki et al. 95], etc.

I.6.3. Motion Computation

Koch [Koch and Li,95] stated that no chip described in the literature reliably solves the problem of motion computation. It is a complex problem to signal speed and direction of motion over a reasonable range of velocities, spatial frequencies and contrasts. Furthermore, reported experiments are done under controlled laboratory conditions. It demonstrates the difficulty of such an important task for practical applications as calculating time-to-contact, segmentation and attention based on motion and so on.

In Neuromorphic Engineering, short range (or intensity) methods are used to compute motion. These methods use image brightness to estimate motion in every pixel so they are noise-prone and present the aperture problem.

The first commercial application of a neuromorphic motion implementation is the control of an optical mouse. Instead of using the mechanical movement of a tracking ball, a vision algorithm was used to perform the same task with no moving parts.

Most implementations use the algorithm method [Delbrück,93]. They subtract the incoming image with a delayed and transposed version of it. Thus, it maximizes output of pixels that $I(x,t)=I(x-vt,t+t)$, that is, pixels that are moving at velocity v. It means that it can only be computed a velocity direction and module at a time.

I.6.4. Segmentation

Although segmentation has not focused as much interest as early vision tasks, some effort has been done in that direction.

Luo et al. [Luo et al,92] published a design that segregates figure from ground. Input was a binary representation of the edges of an image that can be obtained through the design presented in [Bair and Koch,91]. It was a switched resistive network. Each pixel was connected to each of its neighbors through a resistive element and a switch. When there was an edge in the connection, the switch was open and no current flowed through the elements. If there was no edge, a resistive connection was established. Pixels of the periphery were connected to a low voltage (V_{low}) and the center pixel of the network was supposed to belong to the figure and was stuck to a higher voltage (V_{high}) . Then, this higher voltage was 'spread' on the resistive network, thus, pixels that were inside the contour delimited by edges, relaxed to V_{high} and pixels outside the contour relaxed to V_{low}. As connections were established through resistors, small discontinuities of the contour did not degrade the overall performance of the system.

The circuit was implemented on CMOS technology and successful results have been reported. However major drawbacks of this design are that it can only segregate one centered figure and it can not recognize figures with large gaps in their contour.

Synchronization has also been used for segmentation. In 1994, Andreou and Edwards [Andreou and Edwards,94a] [Andreou and Edwards,94b] presented a design, which used the self-resetting neuron [Mead,89a] as basic oscillator. Synchronization was achieved through connecting capacitors of cells that have no edge between them and mismatch easily desynchronized non-connected cells. A one-dimensional version of the model that correctly phase-locked has been implemented. However, results must be extracted from cross-correlation instead of phase coupling because no random synchronization can be guaranteed due to the lack of a global element that breaks away synchrony of non-coupled groups of oscillators.

In addition to these, we have done some work in LEGION implementation. First, we proposed a system that used oscillators very similar to LEGION [Cosp et al.,98] and a sample implementation of a simplified oscillator was presented. Numerical MATLAB and SPICE simulations showed its feasibility. However, this implementation had significant area overhead. Therefore, the costly oscillators were substituted by astable oscillators and applied to a simple synthetic image [Cosp and Madrenas,99a]. This demonstrated that simple electronic oscillators could be used instead of the complicated (in terms of VLSI implementation) oscillators used in LEGION simulations, which are simple to simulate and analyze but difficult to implement using transistors. Further work [Cosp and Madrenas,99b] extracted a higher model of oscillators that allowed us to simulate a bigger network and confirmed that propagation delays are an important limitation for high resolution images.

I.6.5. Neuron Oscillator Implementation

When mimicking a neuron behavior, we must bear in mind its application. If an exact approximation and parameter control are desired for biological research purposes, a large silicon area overhead and power are needed, while if some computing abilities are wanted, the neuron can be simplified to reduce its cost. Although scientific community has done a great effort in modeling neurons, this work is going to focus on simplified models because our aim is to extract computing abilities of neurons.

The first oscillatory electronic artificial neuron was built in 1983 by Keener [Keener,83] and used standard circuitry available at the moment (Op.Amps., discrete resistors and capacitors). Keener designed and implemented a piecewise-linear approximation of the FitzHugh-Nagumo equations and reported successful results. However, this design was focused to a discrete technology and it is not suited for a microelectronic technology.

A piecewise-linear microelectronic implementation of an oscillator is presented in [Rodríguez-Vázquez and Delgado-Restituto,93]. This design is focused on a systematic approach to designing linear systems for modeling chaotic oscillators. Thus, oscillators are too big for a specific application where not so much control on parameters is needed.

A simpler implementation of a neuron is the one presented by Mead in his book on Neuromorphic Engineering [Mead,89a]. Mead's self-resetting neuron is well suited to model an axon circuit but not enough control on its parameters is available for computing purposes treated in this work.

Integrate-and-fire neuron implementations have been designed in [Schultz and Jabri,95]. The design is claimed to behave as ideal integrate-and-fire neurons found in the literature but no practical implementation is given.

The Morris-Lecar model is a neuron model that possesses physiological properties similar to those of actual neurons. This model has been designed and successfully implemented [Patel and DeWeerth,97]. However oscillations are on the order of hundreds of millivolts when power supply is 5V that makes difficult reading of the output. A digital signal (V_{DD}-GND) is desired for computing purposes.

The model that best suits computing purposes for oscillatory computation presented in section 1.5 was presented in [Linares-Barranco et al,89] [Linares-Barranco et al,91]. Various models of oscillatory neurons were presented. Some of them were precise but large. However, others were smaller as the current-mode version of the hysteresis neuron cell. A current mode hysteresis comparator, a nonlinear resistor and an integrator performed oscillations at low cost and hysteresis was easily controlled through input currents. This approach is very useful in oscillatory implementations of neural networks because no additional circuitry is needed to sum input synapses thus considerably simplifying the design. An improved version of this design has demonstrated its computing capabilities in [Cosp and Madrenas,99a].

I.7. THESIS OBJECTIVES AND OUTLINE

The work presented here focuses on the microelectronic implementation of oscillatory neural networks for image segmentation and connected component labeling. The goal is to analyze, implement and study the possibilities of an electronic circuit that is able to segment and label different objects of an image using very large scale integration electronics for low power applications.

- Based on bioinspired computational models, a VLSI oriented algorithm has been proposed and studied analytically and by simulations in Chapter II.
- In Chapter III, a microelectronic design for a specific available technology is presented and their characteristics are checked to accomplish segmentation algorithm conditions.

- Chapter IV presents the experimental results of the implementation of the circuit presented in Chapter III. Its functionality as an image segmentation device is validated as its power requirements.
- Finally, in Chapter V, having explored the possibilities of such networks, this dissertation ends with concluding remarks and presents some future lines of research for neuromorphic segmentation circuits.