BIOLOGICALLY INSPIRED MULTICHANNEL MODELLING OF HUMAN VISUAL PERCEPTUAL SYSTEM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MEHMETCAN APAYDIN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

DECEMBER 2005

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Canan Özgen
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science

Prof. Dr. İsmet Erkmen
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science

Prof. Dr. Aydan Erkmen
Co-Supervisor

Prof. Dr. İsmet Erkmen
Supervisor

Examining committee members:

Assoc.Prof.Dr. A.Aydin Alatan (METU,EEE) _____

Prof.Dr. İsmet Erkmen (METU,EEE) _____

Prof.Dr. Aydan Erkmen (METU,EEE) _____

Assoc.Prof.Dr. Gözde Bozdagi Akar (METU,EEE) _____

Asst.Prof.Dr. İlhan Konukseven (METU,ME) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Surname:

Signature :

ABSTRACT

BIOLOGICALLY INSPIRED MULTICHANNEL MODELLING OF HUMAN VISUAL PERCEPTUAL SYSTEM

APAYDIN, Mehmetcan
M.Sc., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. İsmet Erkmen
Co-supervisor: Prof. Dr. Aydan Erkmen

December 2005, 61 pages

Making a robot autonomous has been a common challenge to be overcome since the very beginning. To be an autonomous system, the robot should collect environmental data, interpret them, and act accordingly. In order to accomplish these, some resource management should be conducted. That is, the resources, which are time, and computation power in our case, should be allocated to more important areas.

Existing researches and approaches, however, are not always human like. Indeed they don't give enough importance on this. Starting from this point of view, the system proposed in this thesis supplies the resource management trying to be more 'human like'. It directs the focus of attention to where higher resolution algorithms are really needed. This 'real need' is determined by the visual features of the scene, and current importance levels (or weight values) of each of these features. As a further attempt, the proposed system is compared with human subjects' characteristics. With unbiased subjects, a set of parameters which resembles a normal human is obtained. Then, in order to see the effect of the guidance, the subjects are asked to concentrate on a single predetermined feature. Finally, an artificial neural network based

learning mechanism is added to learn to mimic a single human or a group of humans.

The system can be used as a preattentive stage module, or some more feature channels can be introduced for better performance in the future.

Keywords: Human-like, vision, perception, visual attention

ÖZ

İNSAN GÖRSEL ALGILAMA SİSTEMİNİN BİYOLOJİK TEMELLİ ÇOK KANALLI MODELLEMESİ

APAYDIN, Mehmetcan
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. İsmet Erkmen
Ortak Tez Yöneticisi: Prof. Dr. Aydan Erkmen

Aralık 2005, 61 sayfa

Bir robotu kendi kendini idare eder hale getirmek, en başından bu yana aşılması güç problemlerden olagelmiştir. Robotun kendini idare eder hale gelmesi için çevresel veriyi toplayıp yorumlayarak ona göre tepki vermesi gerekir. Bunu sağlayabilmek için kaynak yönetimi yapılmalıdır. Yani, zaman ve işlemci gücü gibi kaynakların daha önemli alanlara yönlendirilmesi gerekir.

Ancak halihazırdaki çalışma ve yaklaşımlar her zaman insan benzeri değildir. Ya da buna yeterli önem vermemektedirler. Tezde önerilen sistem bu bakış açısından yola çıkarak anılan kaynak yönetimini insana daha çok benzer olmaya çalışarak gerçekleştirmektedir. İlgi odağını, gerçekten de daha yüksek çözünürlüklü algoritmaları gerektiren yerlere yönlendirir. Bu 'gerçek gereksinim'in konumu ise sahnenin görsel özellikleriyle bu özelliklerin her birine verilen öneme (ya da ağırlığa) göre belirlenir. Daha ileri gidilerek önerilen sistemin insan görsel algı özellikleriyle kıyaslaması yapılmış ve önyargısız deneklerle yapılan deneylerden ortalama bir insana benzer sonuçları üreten ağırlık değiştirgeleri kümesi elde edilmiştir. Daha sonra yönlendirme etkisini görebilmek için deneklerin bir tek görsel özelliğe yoğunlaşmaları istenmiştir. Son olarak yapay sinirsel ağ tabanlı bir öğrenme mekanizması eklenerek sistemin

bir kişi ya da grubu taklit etmeyi öğrenmesi sağlanmıştır.

Sistem, olduğu gibi ele alınarak bir ön-ilgi modülü olarak kullanılabilir ya da daha iyi başarım için daha çok sayıda görsel özellik kanalı eklenebilir.

Anahtar Sözcükler: İnsan benzeri, görme, algılama, görsel ilgi

To My Family and Burcu

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor Prof. Dr. İsmet Erkmen and my co-supervisor Prof. Dr. Aydan Erkmen for their guidance throughout the preparation of this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CCD: Charge Coupled Device

CIE: The International Commission of Illumination

CMOS: Complementary Metal Oxide Semiconductors

HSI: Hue-Saturation-Intensity color space

LMS: Least-Mean-Square

LLS: Longest Line Scanning

NN: Neural Network

OCEM: Occluded Circular Edge Matching

OFE: Optical Flow Equation

RGB: Red-Green-Blue color space

ROI: Region of Interest

USB: Universal Serial Bus

CHAPTER 1

INTRODUCTION

1.1 Motivation

Robot vision systems mainly rely on high resolution imaging technologies, and complicated image processing algorithms. With the increased complexity of the programs, the need for faster computer systems arises.

To reduce this necessity to some extent, some resource management should be conducted. That is, the resources, which are time, and computation power in our case, should be allocated to more important areas.

Once it is known that the human visual system adequately directs the resources to really attractive areas, the solution must be obtained considering the human systems' properties.

Existing researches and approaches are not always human-like. Many times, indeed, the similarity to human visual systems is not given importance. For instance, some works ([2][6]) try to achieve the "context/task dependency" while some others ([21][22]) use biologically inspired sensors but does not take the context into account. In order to get closer to human performance, a visual system should contain as much human properties as possible though. Starting from this point of view, it is aimed to construct a more human-like system throughout the thesis.

1.2 Objective and Goals

The resource management mentioned in the previous section is well done by many animals, especially by human beings, even without conscience. Moreover, there are not much approaches considering the similarity to human. Our first objective is then, to construct a "more human-like" visual perception model such that it is applicable to robots. Secondly, the system should include a learning mechanism to be trained to imitate a single person, or a group of people.

In order to manage this construction, there are some steps that should be overcome.First, a main "scene observer" system should be built. This system is the main part and looks at the "interesting" parts of the scene given some initial information. Then, a measurement of the similarity to human should be performed. In order to measure how similar (or different) the main part behaves like a human, necessary data from the real human subjects should be collected. The learning mechanism then should use this data to learn the characteristics of the subject set. Finally, in order to achieve system flexibility, the topology should be modular. That is, without too much effort one can include or exclude any part of the overall system.

1.3 Methodology

Directing the computing power, or focusing on necessary regions is the main point to achieve. For this achievement, being inspired from the biological systems, features of human visual perceptual system should be imitated. While

constructing this imitative system, it must be noted that in human brain, there are many distinct neuron groups specialized on a specific task such as tasting, smelling, or hearing. These groups deal with their input stimuli individually, and after they are finished, other neural nets takes their output to act upon. Similar to this, the proposed system must take each different visual feature individually, process them, and produce an output for any upper level mechanism. During this process, either parallel computing algorithms or any sequential one may be used with the help of the high speed of current computer systems with respect to any biological neural system. This "feature based" separate processing is applied using different visual features into account[13]. Each visual feature is fed into a channel, and every channel is processed separately[2][6]. After all necessary calculations are done with all of the channels, an upper level channel (i.e. master channel) is formed using the outputs of these primary channels.

How similar is the proposed system to a human visual system? In order to give a satisfying answer to this question, a measurement methodology is to be formed. Since the purpose is not building a real-time interface between a human and the computer but to find the similarities and differences between them, the setup should be able to observe the human only. It is not necessary to interact with the main "scene observing system" or maintain the comfort of the subject. There are some studies to find the gaze point (the point which is under inspection of the eyes at the moment) of a human without any intrusion [11][1]. However, in our case the intrusion may be allowed. The measurement system may be fixed onto the subject's body for a short term intrusion. Indeed, this intrusion is quite useful to further simplify the observing algorithms. Actually, the core algorithms are the same but non-intrusive approaches have additional processes to locate the eye in the captured image. By allowing this short term

intrusion, we get rid of the computational complexity and possible additional errors from these extra processes.

The output of the main system and the data obtained from the measurement system are, if used without each other, useless. They should be interpreted together in some manner and the main system should be adjusted accordingly. This is nothing but learning. The learning mechanism should get the results of the main system and the human subjects, calculate the instant error, and adjust the parameters of the main system to decrease this error value. This task is performed by a three layer feed forward artificial neural network with error backpropagation. In learning applications, this kind of neural networks are widely used because of the stability reasons[16].

1.4 Contribution of the Thesis

This work introduces the human likeliness concept into the visual system. That is, real life data is obtained from human subjects to be compared with the artificial outputs of the system. Moreover, the learning mechanism makes it possible to train the system to imitate a single or a group of people's visual characteristics.

1.5 Outline of the Thesis

The content of this thesis is organized as follows:

Chapter 2 includes the researches and literature about biological visual systems, as well as human inspired computer vision systems. In addition, some

state of the art image processing algorithms are briefly introduced.

In chapter 3, proposed system is described in detail. In the first half of the chapter, the attention system is introduced, and in chapter 4, the human eye tracking system is described.

Chapter 5 describes the simulations, and includes experimental results. The bias effect on human characteristics is also introduced in this chapter.

Chapter 6 gives a summary of the thesis and mentions possible future work.

CHAPTER 2

LITERATURE SURVEY AND BACKGROUND

2.1 Biological Vision

Light is used by many species to get some information about the environment for millions of years, and the most sophisticated and efficient light 'device', the eye, is getting evolved since the first light sensing cell. Today, thanks to evolution, every human being have a pair of these devices to sense light in order to perceive the surroundings.

2.1.1 Eye

Eye is the sensor package of the biological visual system to get the information from the outside by means of light values and to send the obtained information to the processing units. Being quite similar to modern cameras, a lens is subject to the incoming light. The biological lens, unlike the one used in cameras, is elastic, and controlled by a group of muscles for adequate focusing. In front of the lens is the 'Iris' which corresponds to the diaphragm in cameras. The amount of light going towards the receptors is adjusted by this structure. Light then passes through all the eye-sphere and reaches the receptor field, called retina.

2.1.2 Retina

The sensor field of the eye is made up of 130 million separate receptor cells, ganglions and optical neurons. The number of receptors in the eye is about 25 times those in an average 5 megapixels camera. In addition to this overwhelming number, each receptor is nearly 25 times more sensitive than photoreceptors in a camera [7].

When the orientation of the light-sensitive cells is studied, it is seen that there exist two main groups; rods, and cones. The rods cover all of the receiving area on the retina. They are very sensitive to light and to motion. Their mean wavelength of highest sensitivity is between the green and blue region of the electromagnetic spectrum. Although rods cover the entire receptor area, they obtain only an image of low spatial resolution to higher levels of visual-perceptual system. Cones, however, are less sensitive to the intensity of the light relative to the rods, but are used to get the color information (representing red(R), green(G), and blue(B) pigments) of the image. The cones are concentrated in a very small area located on the optical axis called 'Fovea'. A quite high-resolution color image of the central area of the scene is obtained by the use of cones.

2.1.3 Foveal Vision

The high density of light sensitive cells at the center decreases toward the periphery of the sensor field[20], yielding a lower resolution at the periphery of the scene. The use of this low resolution peripheral image is to detect movements or other interesting events such as an area of continuous brightness

fluctuation or a blinking light source etc. When such an event is detected, the oculomotor system redirects the eye so that the region of interest (ROI) is ensured to fall into the Foveal region[25]. This way, the neurons responsible for the visual system are supplied with as much information as possible about the region of interest.

2.2 Robot Vision

When robots tend to achieve full autonomy, the need to percieve the environment rose. Although touch sensors, pieso-sensors, proximity sensors etc. provide a lot of information, 'seeing' the world is the culminating sensing need. A robot's vision system aims to create a model of the real world. This model can be obtained using the knowledge about the objects in the scene, looking angle, and contextual needs etc. To be able to form this model, a robot vision system recovers useful information about a scene from its two dimensional projections(i.e. the images of the scene)[10]. Thus it can be said that robots 'see' their surroundings through the cameras that have been installed on them.

2.2.1 Camera

Like an eye, a camera is used to get the light information , and send it to the processor in an appropriate format. The main difference between a human eye and a camera is at the sensor field. Similar to the cells in an eye, a camera is supplied with photo sensors. The technology used in these sensors may vary: charge coupled device(CCD) or complementary metal oxide semiconductor(CMOS) can be used according to the area of application. The differences

between these technologies are out of the scope of this thesis, however, almost all cameras have a sensor field where the elements are uniformly distributed.

2.2.2 Grid Patterns

The light receptors on a sensor chip are located in such a way that the color order and the shape of a small area is repeated throughout a surface, generating a pattern. This small paterned area can be made up of hexagonal, or generally rectangular light receptors. Thanks to the uniformity of the pattern, all subsections of the acquired image have the same resolution, same color properties and thus carries same amount of information.

In fact, the majority of image processing algorithms rely on this uniformity of the grid patterns. However, this is not the case in a biological eye. The light sensitive cells on the retina are concentrated on the central area. The advantage of this non-uniform distribution is that more bandwidth and processor time can be allocated to areas of interest, avoiding unnecessary data from the other parts.

Log-polar Distribution

One implementation of non-uniform distribution is the log-polar distribution [20]. Light receptors in this distribution type are arranged over coaxial circles. On each circle there are constant number of receptors yielding lower resolution on the outer circles. After the acquired log-polar data is mapped into cartesian coordinates, any local operator used for rectangular grid images can be applied without any adaptations [20]. Although this distribution type is

desirable because of its fidelity to biological cell distributions, manufacturing the necessary solid state log polar sensors limits the application.

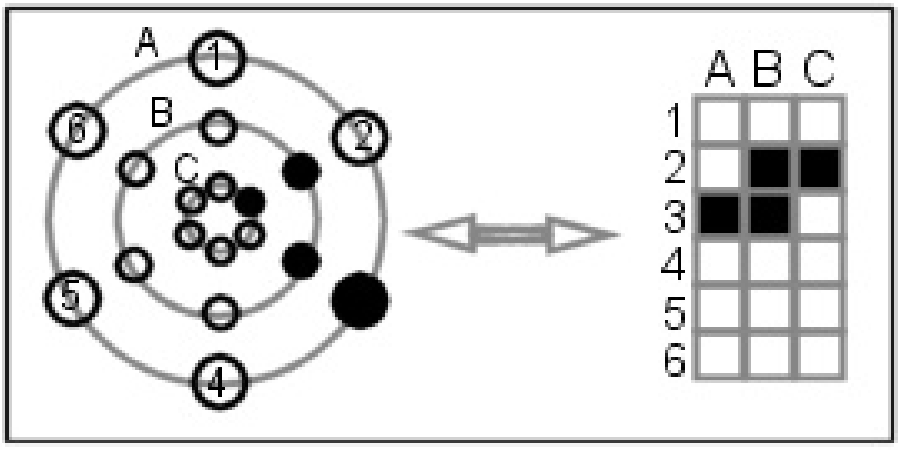

Figure 2.1: Log polar distribution and its cartesian correspondance

Two Camera Approach

Instead of designing and producing space variant sensor fields, it may be preferred that a low resolution wide angle peripheral camera and a coaxially oriented higher resolution narrow angle foveal camera are used for each eye [25],[22]. This allows two separate processors that handle their own cameras and process only the data from the relevant camera. For example, a peripheral processor gets images from the peripheral camera, and detects some movement, sends appropriate signals to the motors to direct the cameras to that point. Then, the second processor that assumes the role of foveal processing performs an object recognition task from a high resolution image taken by the foveal camera.

2.2.3 Fovea Direction

Regardless of the preferred method, the main goal is to get the image of the point of interest by the foveal region. And thus to allocate more bandwidth and processor time to the region of interest. This task can be performed by first determining the area of interest in the peripheral scale. After that, the cameras can be rotated to the calculated angle. This rotation, mimicking the human visual system, should have at least two degrees of freedom. Assigning x-axis to be parallel to the line joining the two eyes, and z-axis to be the vertical one, the cameras can be directed to any point by rotation about these two axes.

Although such a sequence helps to mimic human eye movements, this hardware implementation has also some disadvantages. The cameras and related circuits have a mass that should be moved by fast actuators. The circuitry adds additional delay to that caused by the inertia of this mass. In order to have faster response, software approaches can be used instead of, or combined with, the hardware direction. That is, a single camera is used for obtaining both peripheral and foveal images. First, the peripheral image is processed to determine the region of interest. Then, further high-resolution processes can be performed on the region of interest, optionally the hardware actuators can be started to operate to catch up with scene changes. The location changes in the region of interest are detected, and the most processor power is directed to that area. This way, the response time is quite small compared to the hardware rotation approach since there is no physical body to move, or accelerate. The disadvantage of this technique is that some bandwidth is sacrificed for using the same camera for both peripheral and foveal images.

2.3 Image Processing

2.3.1 Color Image

The light captured by either an eye or a camera has a distribution of different wavelengths generating the color, together with an intensity value. Although the wavelength and intensity values are naturally continuous, 'a practical image system, including the human vision system, works with a small number of samples from the distribution of wavelengths' [10]. This enables us to represent colors in a finite set.

There exist many well-developed sampled spaces to represent a specific color. One of them is the standard chromaticity diagram published by the CIE (the International Commission on Illumination)[18].

CIE defines any specific color by two chromaticity values x,y, and an intensity value Y. The meaningful, or perceivable, region on x-y plane forms a triangle-like region on the diagram. At around each corner of this shape, one of the main colors (red, green, and blue) is located.

Hue, Saturation, Intensity

A more 'human-friendly' space can easily be obtained using and simplifying the CIE diagram. Humans perceive light not as numbers but as its color (the main wavelength, or Hue), how colorful it is (saturation) and as its brightness (intensity, or lightness). The hue-saturation-lightness (or hue-saturation intensity) (HSI) space assumes the CIE diagram as a perfect triangle with

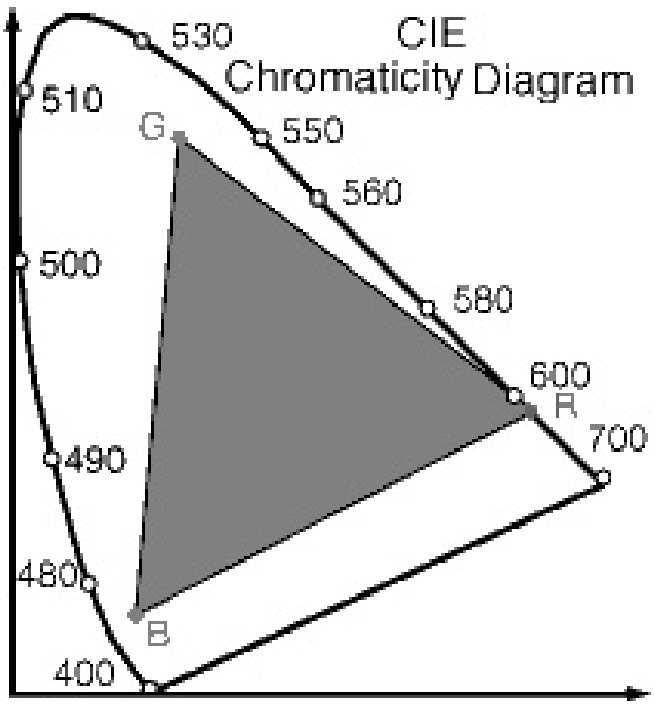

Figure 2.2: CIE Chromaticity Diagram and RGB triangle

a primary color at each corner. To determine any particular color, a line is drawn from the center of the triangle to the point of that color. The angle of that line corresponds to the hue value, and its length to saturation. As HSI definition directly relates HSI levels to RGB values using the triangle, a color described in any model can be converted to the other color space using the following formulae;

$I = \frac{(R+G+B)}{3}$,

$S = 1 - \frac{3}{R+G+B} \min(R, G, B)$,

$H = \arccos \frac{2R-G-B}{2\sqrt{(R-G)^2+(R-B)(G-B)}}$

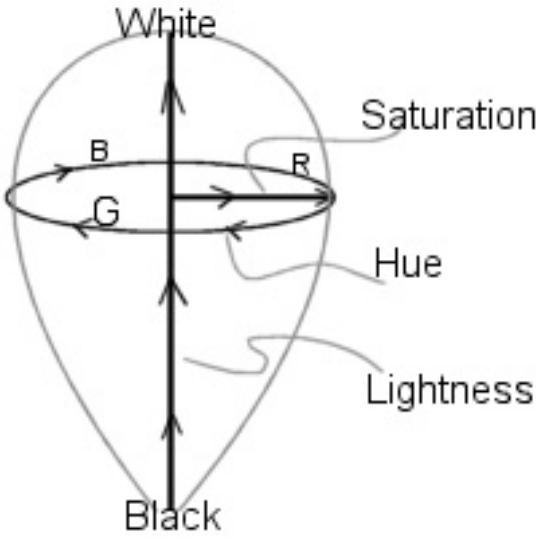

Figure 2.3: Physical representation of HSL space

Distance to the nearest Gray

Although saturation gives a measure of how colorful the light is, another simple conversion may also be helpful. That is, if the specific color's RGB decomposition values are r, g, and b respectively, the gray value s of that color is;

$t = (0.33r + 0.33g + 0.33b)$,

and the value of 'colorfulness' is;

$c = \sqrt{(r-t)^2 + (g-t)^2 + (b-t)^2}$

which is nothing but the Euclidian distance between the color to its gray level representation.

The value of c corresponds to how far is the given color to the nearest gray, or how "less gray" it is. In the three dimensional RGB space, the diagonal line passing through origin and (1,1,1) is the "gray" line. The pixels having the color values on that line are considered to be gray. Any other point has

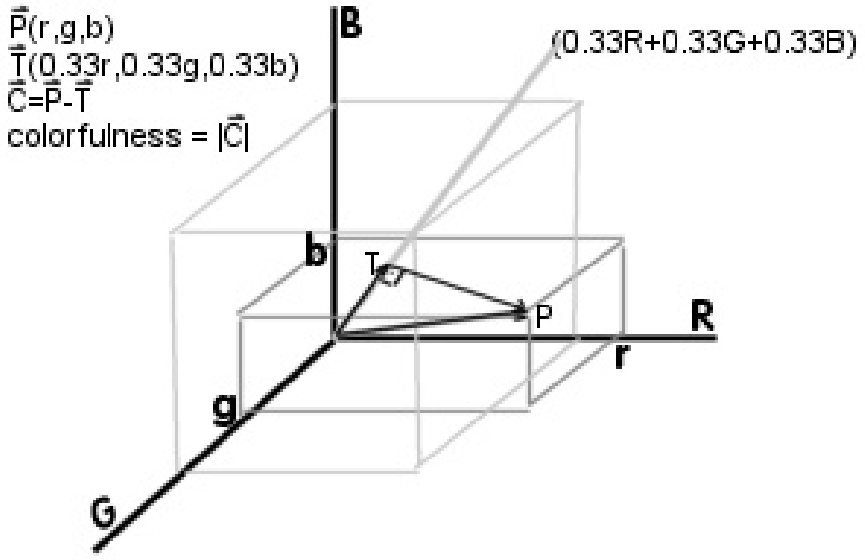

Figure 2.4: Vectorial representation in RGB space

a nonzero distance to this line. The "colorfulness" in our case is just this distance to the "gray" line.

At this point it should be noted that the gray value of a pixel is $t = (0.299r + 0.587g + 0.114b)$ according to CIE. However, taking the coefficients as 0.33 did not cause considerable changes in our case. Namely, a difference of 0.03 is observed which is ignored in the rest of the work.

2.3.2 Motion

Robots have to detect and interpret the changes in the dynamically changing world surrounding them. In order to accomplish this, a robot should be able to detect the differences in consecutive video frames obtained through its vision system. During this frame by frame differentiating job, only the gray levels of the pixels are taken into account as this segregation or detection of the motion

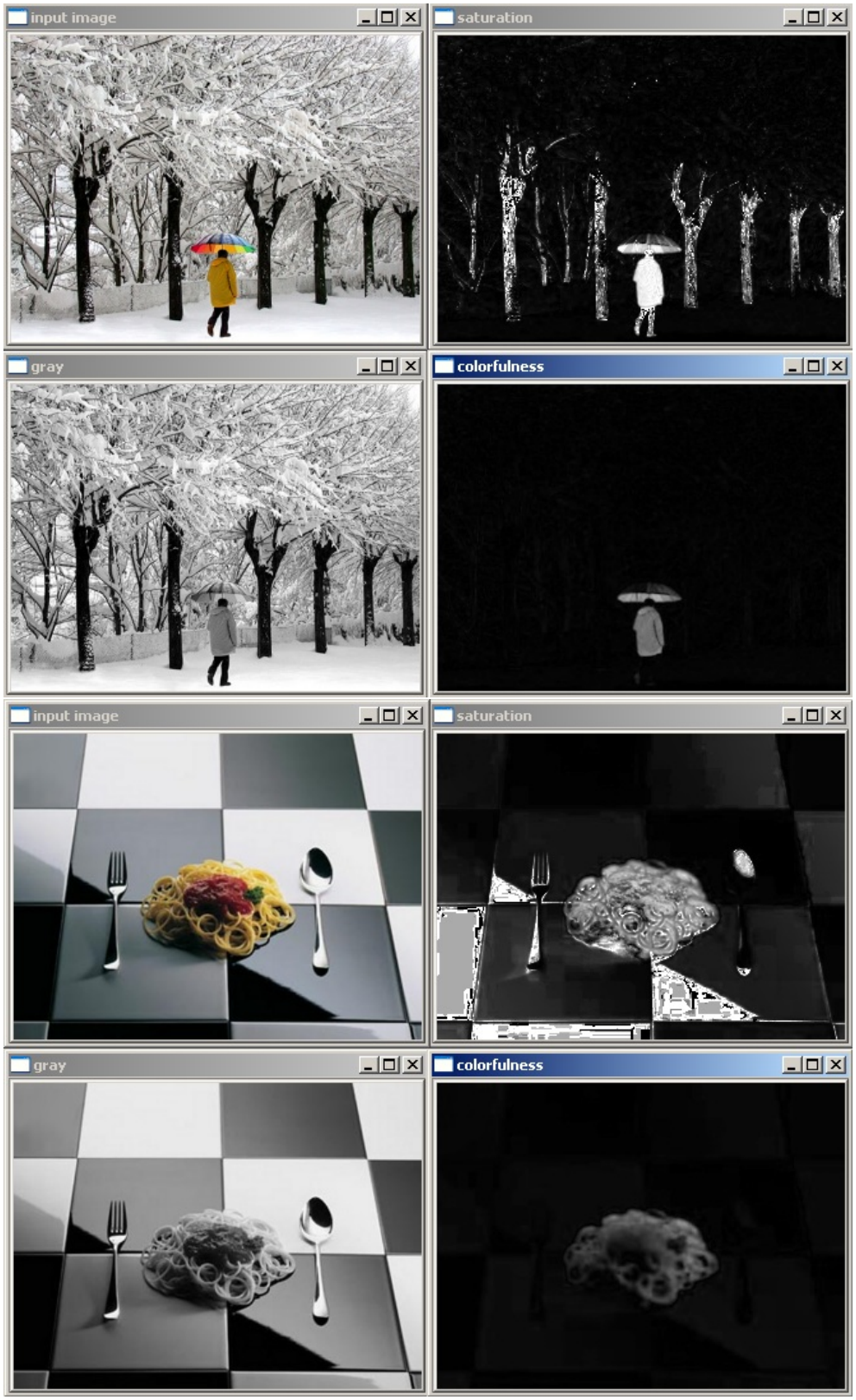

Figure 2.5: Distance to the nearest gray vs. Saturation value of two photographic images. Note that saturation value produces unwanted noise in dark regions.

appears to be color-blind [5].

Motion by Frame Difference

The most useful, and hence widely used, change detection technique is simple subtraction. The change between the frames is then calculated by obtaining the absolute value difference of each corresponding pixel pair from each frames. That is;

$$r(x,y) = \begin{cases} 1 & if|F1(x,y) - F2(x,y)| > \tau \\ 0 & otherwise \end{cases}$$

where F1,F2 are functions of intensities of consecutive frames, and τ is a threshold level.

If there are very small changes in the scene, a smaller threshold level would be required for detection. However, small threshold would cause the noisy points to be falsely detected as motion. To avoid these noisy pixels, size filter can be applied. Size filter simply discards any 4 or 8 connected pre-detected motion pixels if the area of the connected region is below a threshold. This filter consumes extra processing time and may lose some small moving pixels.

Rather than this approach, the motion can be detected within image blocks of size mxn. In each block, pair differences of each pixels are calculated and the block is signed as 'moving', or '1' if the sum of its cells is greater than a threshold.

$$r_{i,j}(x,y) = \begin{cases} 1 & if \sum_{m,n} r(x_{b_{i,j}}, y_{b_{i,j}}) > \tau \\ 0 & otherwise \end{cases}$$

where i,j shows the block at ith column jth row, $x_{b_{i,j}}, y_{b_{i,j}}$ the points within the

block b at column i, row j, and τ being the threshold level.

In our work, since weighted averages are used as will be discussed later, two threshold levels are used. The values falling between these two thresholds are taken as $\frac{1}{2}$ and values larger than the bigger threshold are taken as 1. That is

$$r_{i,j}(x,y) = \begin{cases} 1 & if \sum_{m,n} r(x_{b_{i,j}}, y_{b_{i,j}}) > \tau_2 \\ 1/2 & if \sum_{m,n} r(x_{b_{i,j}}, y_{b_{i,j}}) > \tau_1 \\ 0 & otherwise \end{cases}$$

This way, smaller changes are also taken into account by a factor less than 1, not losing small moving objects, and avoiding any noise-dependant false detections to negatively effect the results.

Motion Vectors

The motion between frames can also be calculated using optical flow methods. Optical flow methods take an equation called optical flow equation (OFE) as the basis. What this equation implies is that in a continuous and differentiable spatio-temporal space, a moving objects' illumination levels (hence the value of corresponding pixels) stay constant. That is, for $s(x, y, t)$ representing the gray value of pixels at (x, y) of the frames at time t,

$$\frac{ds(x,y,t)}{dt} = 0$$

Using chain rule we have an equation called "Optical Flow Equation(OFE)":

$$\frac{\partial s}{\partial x} v_x + \frac{\partial s}{\partial y} v_y + \frac{\partial s}{\partial t} = 0$$

where $v_x = dx/dt$ and $v_y = dy/dt$

As one can notice, the OFE is a scalar equation with two unknowns v_x, v_y. In order to obtain a solution, another constraint must also be introduced. There exist several methods for this, including the well known Horn and Schunck's method [4].

Horn and Schunck's method utilizes OFE with the assumption that the variation within the neighboring optical flow vectors is to be minimum.Let

$$\varepsilon_{ofe} = \frac{\partial s}{\partial x} v_x + \frac{\partial s}{\partial y} v_y + \frac{\partial s}{\partial t}$$

represents the optical flow error. Note that if the optical flow equation is satisfied, ε_{ofe} will be 0. In real life, though, because of noise and occlusion, this hypothetical result cannot be achieved. Instead, ε_{ofe} is minimized. If we also define

$$\varepsilon_s^2 = \frac{\partial v_x}{\partial x}^2 + \frac{\partial v_x}{\partial y}^2 + \frac{\partial v_y}{\partial x}^2 + \frac{\partial v_y}{\partial y}^2$$

to denote the magnitude square of the pixel to pixel change of the velocity vectors v_x, v_y, Horn and Schunck's method introduces the constraint to minimize ε_s. The mathematical overall representation of the method is then;

$$\min \int_{img} (\varepsilon_{ofe}^2 + \lambda^2 \varepsilon_s^2)$$

where λ is the coefficient to adjust the weight of Horn and Schunck's constraint in the overall calculations. Increasing the lambda value, one can increase the influence of the smoothness constraint. However, this value is usually selected heuristically to match the current necessities of the application.

2.4 Human Eye Gaze Determination

When people look at something, they direct their gaze onto that object. This is accomplished by turning the joints at waist and neck, and turning the eye-globes around their x and z axes. By actively tracking the gaze of a human, a robot can be instructed what to pick, where to go etc. There are several techniques to track the human eye gaze. Head movement measurement, Purkinje Image Tracking, Contact Lens Method, Corneal and Pupil Reflection Relationship Method, Electro-Oculography, Limbus, Pupil and Eyelid Tracking are some of these [11]. Many of these are not suitable to be used in a camera-only system. For example, contact lens method requires special lenses and cameras for a specific wavelength. These special lenses reflect the light of a specific wavelength, and with the cameras tuned to that specific wavelength, the exact locations of the pupils are grabbed without any more image processing like color correction or thresholding. Corneal and Pupil reflection relationship method again requires special camera for a wavelength to determine the pupil investigating the differences in the reflective properties of the pupil and other parts of the eye. In this type of methods, biological reflection properties of iris and retina make it possible to locate the pupil. That is, for specific wavelengths, human eye reflects the light like the cats' eyes. The obtained images then contain bright points representing the pupil.

2.4.1 Non Intrusive Methods

If an interface between a human and a robot is to be built and used in real life, it must not disturb the human's life area, or movements. That is, it must

obtain the data from its instructor without touching him, indeed from an adequate distance where he/she is still comfortable. In order to achieve this, the robot must first detect the face. Later, it must locate the eyes, and then find their sight direction. Knowing the orientation of the face and two eyes, the point of interest can be calculated [1]. Taking the orientation of the face as the reference coordinate frame, the eyes' orientations are two non-parallel lines on a single plane. It is trivial that these two lines intersect at the point where the object under interest is located.

2.4.2 Fixed Head

When the main goal is to build an interface between robot and human, non intrusive methods (which do not touch or disturb the human subject) can be preferred. However, for measurement purposes (i.e. not for commanding a robot with eye movements, or in cases where the object should not be aware of the surveillance), it is not so necessary with its high processing time need. Using a fixed head technique, the need for face detection and finding the eyes is avoided, lowering processing complexity. Furthermore, if the location and position of scene is well known, it gets easier to find the point of interest more accurately.

Occluded Circular Edge Matching(OCEM)

In this method, the position of the pupil is estimated from the left and right boundaries of the visible portion of the iris [11]. Top and bottom edges are not being used since they are covered by the eyelids most of the time, hence not visible.

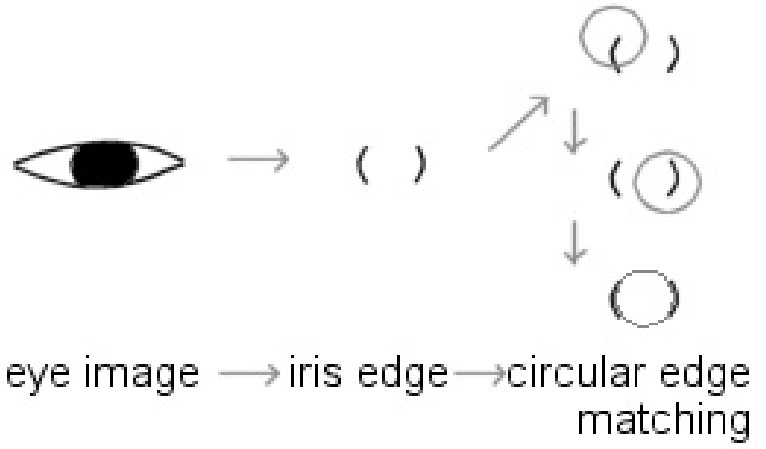

Figure 2.6: Occluded Circular Edge Matching Method (OCEM)

The edge pixels of the iris are found by any appropriate edge detection algorithms (prewitt, sobel, canny etc.). Then, an initial seed point is selected. This selection can be based on centroid, or midpoint of the rectangle that holds every edge pixel within its area. Then, starting from this seed point, the matching step takes place.

A match kernel is produced by modelling the iris boundary as a circle. The match kernel is moved around the seed point a few pixels in all four main directions and at each new location, a score is given to the closeness of the model and the real edge. The point having the highest score is decided to be the center of the iris. A good seed point selection improves the performance as well as appropriate selection of the amount of shift of the kernel. As a disadvantage, the technique requires a clean iris-edge figure without other noisy pixels etc. and this is not easily obtained by ordinary cameras.

Longest Line Scanning(LLS)

This method assumes that the iris has a circular shape and is based on the fact that the center of a circle is at the midpoint of the longest horizontal line within the boundary of that circle. Therefore, the boundary of iris is to be determined first.

To get the boundary pixels, any edge detection methods can be used, but, because of noise and false-edge pixels, we did not opt to use this method. Instead, we applied simple segmentation using a threshold. At this point, another problem should be overcome; the color of the eyes.

Green-Blue Eye Correction

Since both the edge detection and threshold mechanisms work on gray-scale images, the acquired image of the eye should be converted into gray space. During this step, green and blue eyes will have a lighter gray value than those of dark colors.

To overcome this, eye images are investigated under different color spaces (see fig. 2.7) and it is found appropriate to use the R channel as the base. This selection enables the algorithms to deal with darker iris without effecting the already-dark colored eyes' performance(see fig. 2.8).

After the correction and determining the iris pixels, LLS algorithm takes place. The algorithm is as follows:

```
begin
find the centroid of the iris pixels (starting point)
go up and down to measure horizontal lines
```

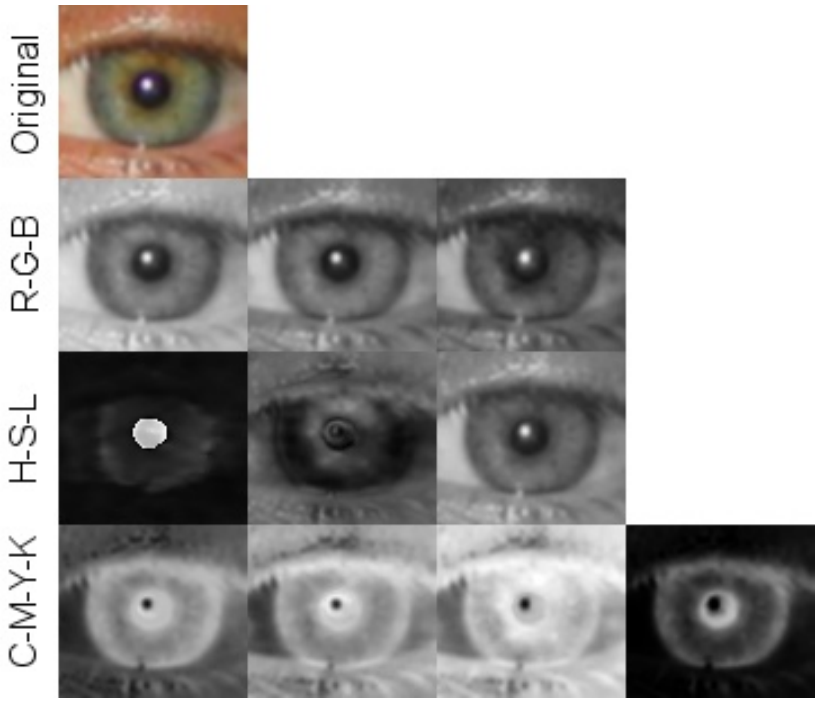

Figure 2.7: green eye image in different color spaces

Figure 2.8: R channel achieves good performance for both green and dark eyes.

```
....until the measure starts so decrease
find maximum measurement
if more than one longest line
....y is the mid-vertical position
else y is the longest line position
x is midpoint of the line at y
return x,y
end
```

LLS algorithm is fast, and simple. However, after experiments, we decided to modify it slightly to get rid of the noisy pixels around the iris. Our algorithm measures the difference between leftmost and rightmost detected pixels. Therefore, if one (or very unluckily both) of these ends are not really iris edges but noise, then the midpoint between them may or may not be the correct center. To avoid this, a second iteration is implemented to start at the midpoint of these suspicious pixels and to find the real center point.

2.5 Visual Perception

The visual processing of a scene can be investigated in three different categories [13], namely vision before attention, vision with attention, and vision after attention. Vision after attention (post-attentive vision) is kind of a separate concept as the post-attentive processes get into the area of cognition. Since all the attentive steps are already accomplished, after that point, cognitive algorithms should be brought onto stage. The robot is now focused on the object of interest. However, only one object or region can be kept under attention. Therefore, when the system is focused on a single part, it is possible to change the appearance and/or other properties of an object that fall outside of the already attended region [14]. A repeating preattentive search mechanism

is compulsory to avoid this 'change blindness' [14]. The preattentive vision, on the other hand, deals with the overall scene, and processes everything at once.

2.5.1 Preattentive Vision

Preattentive vision can be summarized as to find, or at least direct the spotlight of attention onto a desired region in the scene. For example, think of a system to search a metallic needle among a hundred of toothpicks which are of the same size and shape of the needle. The past experiments dictated that a metallic object is 'shiny' under general lighting conditions. So, the preattentive processes are initiated to find regions whose distinctive feature is being 'shiny'. After the processes finish, the system then focuses only on the candidates instead of progressively searching all over the image. Besides being 'shiny' or 'matte', many preattentive features can be found.

Preattentive Features

Although many other features can be thought of, the ones discussed in [13] form a good basis. These features will briefly be explained in this section.

Color

Color is a preattentive feature which is efficient when the objects in the scene are not too similar in color. Color in computer world has three dimensions regardless of which color space is considered (RGB, HSL, YUV etc.). Therefore, if this 3-D space can be separated into segments that include only one color

but not the other, any color can be selected among the regions dominated by another one.

If widely separated colors are considered, the distinction may be more easily recognized computationally when the color space can be separated into pieces each having the nearest color to the so called 'main' colors which are red, green, and blue.

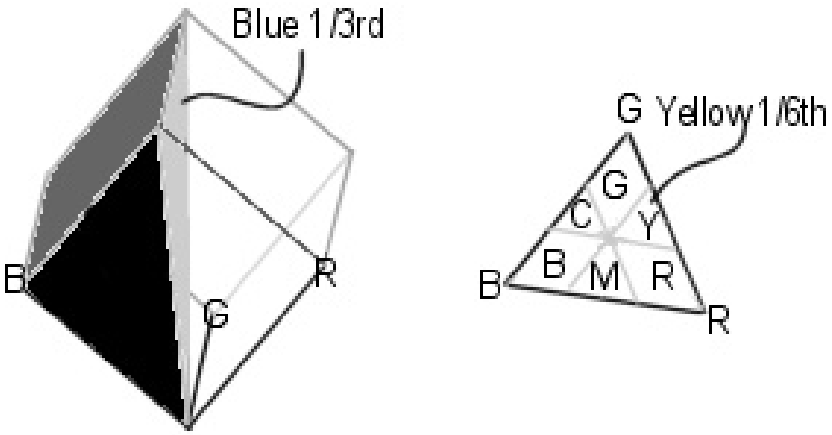

Figure 2.9: Simplest color segmentation using main colors

As an alternative, a six segment space can be obtained introducing the main secondary colors cyan, magenta, and yellow.

Orientation

Orientation of an object is defined as the axis of elongation [10]. The axis of elongation can be found by minimizing the sum of square perpendicular distances of each object point to the line. That is, the axis is the line which

minimizes D;

$$D = \sum_{i,j} d_{ij}^2 . B(i,j) \tag{2.1}$$

where d_{ij} is the perpendicular distance of point (i,j) to the line, and

$$B(i,j) = \begin{cases} 1 & if point(i,j) \in object \\ 0 & otherwise \end{cases}$$

The orientation of an object is useful if it is different enough from the other objects' orientations. Moreover it is more efficient when the 'other' objects' orientations are more homogeneous [13].

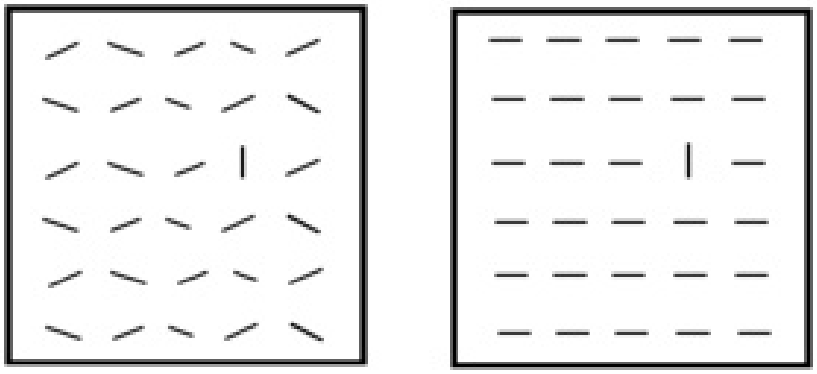

Figure 2.10: Orientation is more efficient if 'other' objects' orientations are more homogeneous [13]

Motion

Motion of an object can be detected by differentiating two consecutive frames. This method may yield erroneous results if there is a large global motion. However, as long as the preattentive and peripheral view is under consideration

the global motion can be assumed out of context.

The preattentive feature motion is very efficient under no global motion. This is most probably due to the fact that attention is attracted to the points of local change [13].

There are other researches on motion especially considering the optical flow. An option to select between frame difference and vectorial motion is included in the system. However, it it seen that vectorial motion calculation does not yield much enough improvement compared with its time-doubling computing load.

Depth Fields

Preattentive vision must also take the third dimension into account. The depth fields of different regions on a 2-D image definitely improves the preattentive search. The third dimension is obtained processing the disparity (the amount of the shift of the locations of an object on two eyes' obtained images), and used to get the attention to higher disparity points. However, like any other 3D features, it requires a second image, a stereo pair of images to process.

Lustre

Lustre is produced by putting a bright field on one side of a stereo pair while putting a dark one at the corresponding point on the other side. The existence of lustre makes the surface appear to be shining. Although shininess can be used as a preattentive measure, this feature, again, is available only for stereo pairs obtained using two cameras.

2.5.2 Vision With Attention

The preattentive vision system, dealing with peripheral image of the scene, is to find the points of interest and to make the attentive procedures focus only on the necessary parts of the image.

The attentive procedures like necessary pattern recognition routines use the information coming from the preattentive layer, and then process the foveal image if available. Depending on the context or the current task, these procedures may find a human face, recognize it, or may detect a recent change on the surface of the object.

After supplying these background information, our proposed system can now be constructed.

CHAPTER 3

PROPOSED SYSTEM

In general, people walk around without thinking of what they see at that moment. However, if someone wants to describe what he is interested in a real world scene, or wants to get the attention of others to his point of interest, the description will most probably be a verbal one like 'the man with red shirt standing near that building', or 'the girl waving a colorful scarf'. Although these descriptions do not include too much information, a normal human will most probably find the described target even in a crowd. More than that, a human (or any animal, in general) will look at a specific point in the scene which is the most attractive one. For example, a yellow and black colored fast moving object, namely a cheetah, will be the most attractive object in the scene for the impala. Again, for a pilot in the cockpit, the most attractive point will be the moving object at the horizon, regardless of its color, or direction.

In the examples mentioned above, one common point is that some preattentive processes like detection of color and motion are used to direct the attention. Our work also takes this point as a basis.

3.1 The Scene Observer

As mentioned before, preattentive vision can be based the preattentive features. Since we use a single camera, the features requiring stereo pair of images

coming from two cameras like lustre and depth fields are not applicable in our work. Actually, the major preattentive feature we considered is motion.

3.1.1 Motion

Motion frame is calculated by differentiating the values of a given pixel in two successive frames. Since the change direction is not important, the absolute values of the differences are appropriate to use. After taking differences, the image is divided into square blocks of size 10x10. Motion detection is applied into these image blocks as;

$$r_{i,j}(x,y) = \begin{cases} 1 & if \sum r(x_{b_{i,j}}, y_{b_{i,j}}) > \tau_H \\ 0.5 & if \sum r(x_{b_{i,j}}, y_{b_{i,j}}) > \tau_L \\ 0 & otherwise \end{cases}$$

where i,j shows the block at ith column jth row, $x_{b_{i,j}}, y_{b_{i,j}}$ the points within the block b at column i, row j, τ_H and τ_L being the higher and lower threshold levels.

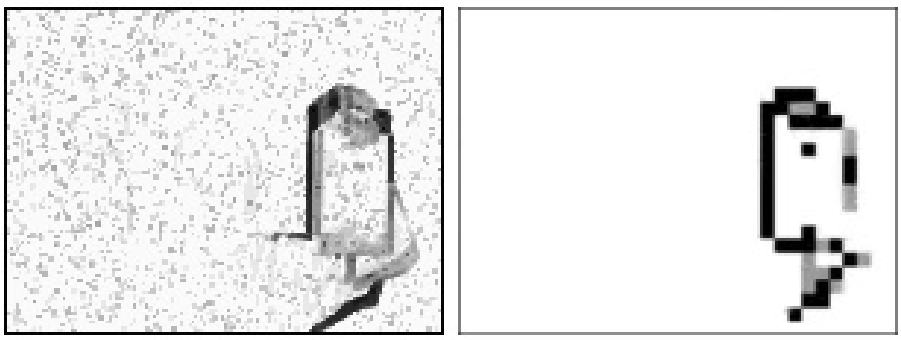

Figure 3.1: 10x10 blocks in frame difference detection provide a noiseless result

After that, the blocks are filled with value 1 for high motion areas, and 0.5 for

low motion ones.

As a second way, the motion frame can be calculated from the magnitudes of motion vector fields. That is, the standard optical flow calculation methods like horn and shunck are applied onto the frame. The magnitudes of the resulting vectors are direct measures of the amount of motion at the corresponding pixel or block of the image.

3.1.2 Color

Although the multi channel approach allows the use of a single color thresholding process for each main colors R,G,B, a unique 'colorfulness' channel is used in our work. All color information is put into a single channel to reduce the computation time. The color channel is built as putting the 'euclidian distance to gray' levels of the pixel colors into the corresponding pixel in the channel, and then applying simple thresholding onto that channel.See section 2.3.1 and fig. 2.4 for the discussion on this 'colorfulness' concept.

3.1.3 Habituation

Habituation can be thought of a preattentive feature which is changing its characteristics as a function of time. When a human subject notices an interesting object, he looks at it. But after some time, he will get used to that, and will get easily distracted looking at other points in the scene other than the previously focused object. In order to model such a characteristic behavior as a feature of the robotic device, a channel called 'habituation channel' is introduced into the system.

Such a model is generated based on the desiderata to have at early times the weight of the initial focal point to be high, and focus to be persistent. As time passes, the model should relax the persistence of the focus, encouraging the visual system to 'look' at other points in the scene rather than sticking onto the point previously under focus.

This is obtained by creating a two dimensional gaussian curve centering at the point of interest. That is,

$G(x, y, t) = F(t).\frac{1}{2\pi\sigma^2}e^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}}$ and

$F(t) = a.max(-1, 1 - \frac{\Delta t}{\tau})$

where (x_0, y_0) is the point of interest, a is the peak amplitude of the function, $\tau, \Delta t$ are the time constant, and the time elapsed since the last reset, respectively.

A habituation reset is needed in order to confine distraction to a limited time, and to regain a certain focus. Habituation reset is done whenever the focus of attention is distracted, and the point of interest moves considerably fast between two consecutive frames. After a reset, the system again will gain some kind of inertia around the center of interest, generating a refocusing. If enough time passes without any habituation resets, the habituation function is minimum at the center, and maximum at the off-center area, the system will be more likely to look at an off-center object in the scene having total distraction out of the scope of the focus.

3.1.4 Linear Combination

For the robots visual system to determine a point of interest, it should take all the previously mentioned preattentive feature channels into account. According to the instant interests or the context, each channel should have different importance. Rather than getting each channels outputs and calculating the point of interest for each frame, it is more practical to combine these channels into a single channel taking their individual importance values into account. In order to achieve this single channel showing the most featured regions, all of the channels (in our work, color, motion, and habituation channels) are summed up in a linear manner as,

$$F(x,y) = w_{color}.C(x,y) + w_{motion}.M(x,y) + w_{habit}.H(x,y) \qquad (3.1)$$

for all $x, y \in image$, where w_{color}, w_{motion}, and w_{habit} are weight parameters of the feature channels C(color), M(motion), and H(habituation) respectively. In addition, in order to avoid clipping in the combined map, the weights are

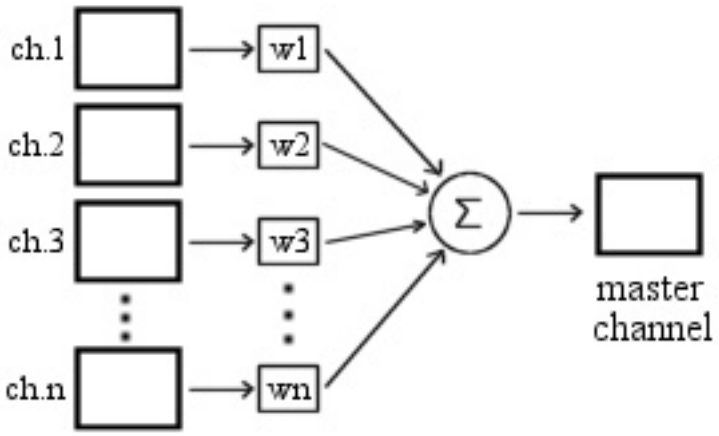

Figure 3.2: Linear combination of feature channels.

limited to sum up to 1. Otherwise, thinking of the worst case where $C(x, y) = M(x, y) = C(x, y) = 1$, $F(x, y)$ would be $(w_c + w_m + w_h) > 1$, which should lie between 0 and 1 in our case. In this case, either this value is clipped and assumed as 1, or the overall image is reprocessed to be normalized at a value of 1. Neither of the cases is efficient. Instead, a precaution to keep the sum of the weights below 1 should be taken.

3.1.5 Interest Point Extraction

The channels are formed and then linearly combined in order to direct the attention of the system to a focus point. This point of interest is determined by the scene observer as finding the center of gravity of the combined image(i.e. the master channel).

Center of gravity of a 2D shape is found as follows: First, the horizontal and vertical projections of the image are taken. For the horizontal projection, an array of the same width as the image is filled with the sum of all pixel values at the image column corresponding to the array cell. Same procedure is applied for the vertical array. Then, for each array, the centroid which is a point where the sum of the weights at the right hand side of it is equal to the sum of the weights at the left hand side is found(see fig. 3.3).

The found values that corresponds to the x and y coordinates of the centroid are taken as the respective x and y component of the interest point. Interest point extraction process also looks at the distance between the last center-of-interest and the newly found point to determine if there is need to reset the habituation process, and warns the habituation process if necessary.

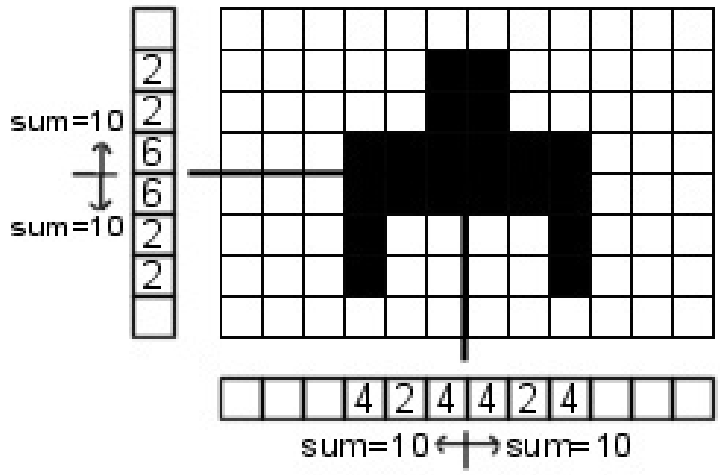

Figure 3.3: center of gravity can be found by its horizontal and vertical components

Figure 3.4: center of interest marked with a high contrast rectangle

As the system has a graphical user interface, the center of interest is marked with a surrounding rectangle, representing the area of interest, or the area supposed to be processed later by the foveal processed, on the screen (see fig.3.4).

3.1.6 Taking Advantage of Multi Layers

The major advantage of the use of the multi channel topology is the system's flexibility. That is, the system can be easily modified to have more or less channels of different visual properties. The only thing that should not be avoided is that the weights of all channels must sum up to 1. To emphasize this flexibility, a "corner detection" channel is included as an example in our system.

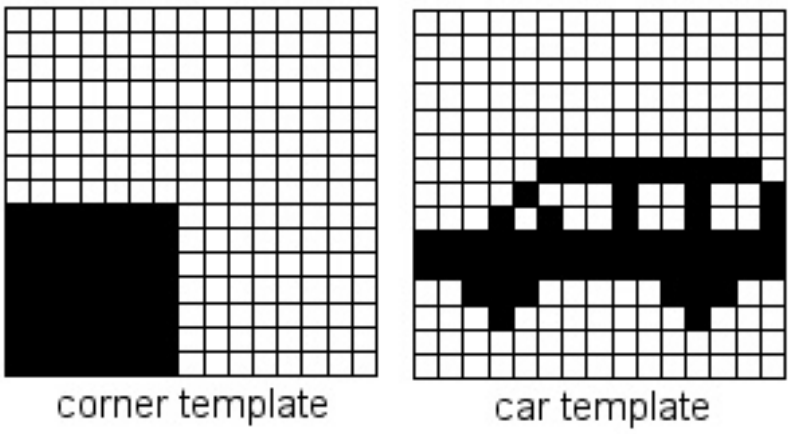

Figure 3.5: A corner template. Expansion with other features is also possible

The corner channel uses a 15x15 corner template as a kernel and convolves this kernel with the overall frame. The corner template contains a 7x7 black region on its lower-left side, and white elsewhere. As a result, the most "corner-like" regions of the frame are found. This corner channel is then fed to the already built linear combination mechanism.

The corner example is chosen because there really are some corners in the used test video sequence, especially the corners of the books. However, this does not mean that one can not use a "smiley" or "car" template of any size.

CHAPTER 4

HUMAN DATA ACQUISITION

4.1 Human Eye Interface

One of our goals in this thesis is to determine how close a computer graphics system can manage attention directing to a human being. Therefore, a human interface is introduced to measure real human characteristics of eye movement.

4.1.1 Camera & Lens

For taking the images of human eye, a camera is needed. There are different types of cameras like composite video, USB, or having own special PCI card to be installed onto the PC. In order to be able to use the system in a variety of computers with ease of installation, USB option is preferred. This way, any USB compliant camera can be used to capture the eye images and any PC with a USB port can be used without installing a video capture card in it.

The camera used in our system is JTech©USB webcam WC3000, which is originally designed as a web cam. However, factory manufactured body is disassembled, and put in another enclosure. At the other end of this enclosure, a glass lens of 50mm focal length is fixed. The use of this lens is to maintain adequate amount of optical zoom. Actually, only the eye occupy the most of the scene area by the use of this optical zoom. The more detailed movement

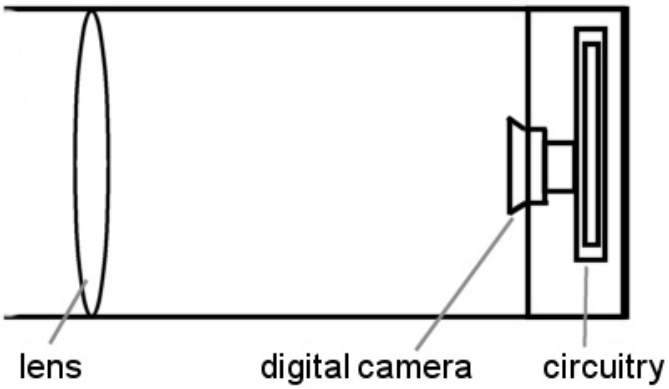

Figure 4.1: Camera and zoom lens in the enclosure.

analysis is wanted, the more optical zoom is required.

4.1.2 The Human Eye Tracker

Although there are non-intrusive approaches to track the eye-gaze of a human, they are not necessary in our case. A simpler fixed-head method would give appropriate results. Furthermore, if the fixed-head is also fixed to the scene, the task of finding the place at the gaze point gets simpler.

For this purpose, a human eye tracking device is designed to be mounted on the monitor of the PC. At the user side, the human subject looks at the monitor through a circular aperture of diameter 4cm while fixating his head to the apparatus. At the other side stands the PC monitor and just below the monitor there is the camera looking directly at the eye aperture. While the human subject watches the PC monitor, the camera takes images of the eye in different gaze positions.

Another property of the apparatus is to block ambient light to pass into the camera as it is made up of opaque material. The changes in the total amount

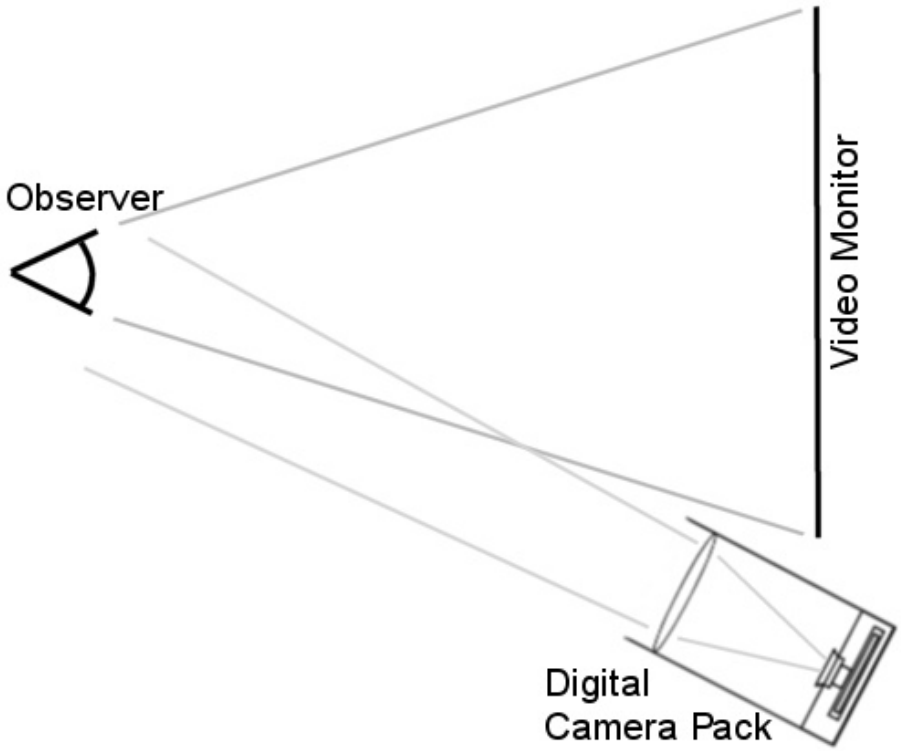

Figure 4.2: Sketch showing the apparatus.

of light going into the camera causes the automatic white balance system of the camera to operate, which must be avoided in order for the algorithms to work properly.

4.1.3 Eye-gaze Detection

Iris Center Location

In order to find the center point of the iris, green-blue eye correction algorithm is applied onto the taken eye image first. This algorithm takes the R channel of the eye image as the original eye image is not ready to be processed if the subject's eye is green or blue. After the correction, thresholding and deter-

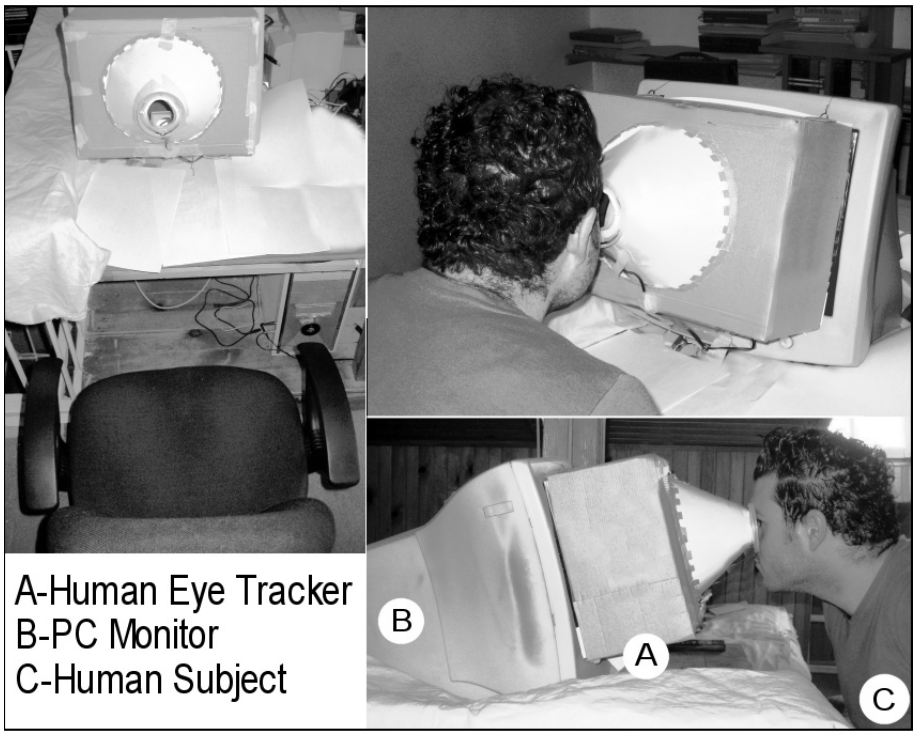

Figure 4.3: Human eye tracker experimental setup

mining the iris pixels, LLS algorithm takes place.(see section 2.4.2) The LLS algorithm returns the x and y components of the pixel at the center of the iris, or pupil.

Calibration

At the beginning, the system shows a sequence of images on the monitor to be in calibre with the current subject watching the screen(see fig.4.4). These images are plain black, with a bright yellow sign at the very corner of each one. Each image has the yellow sign at a different corner. This way, the eye of the human subject is forced to look at the four edges of the video frame.

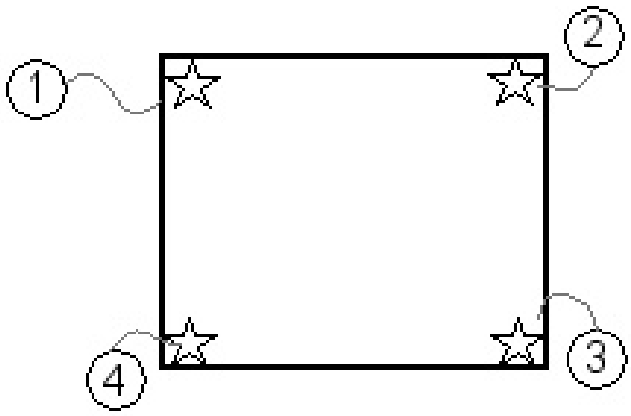

Figure 4.4: Calibration of Human Eye Tracker: A bright sign is shown at the corners in sequence

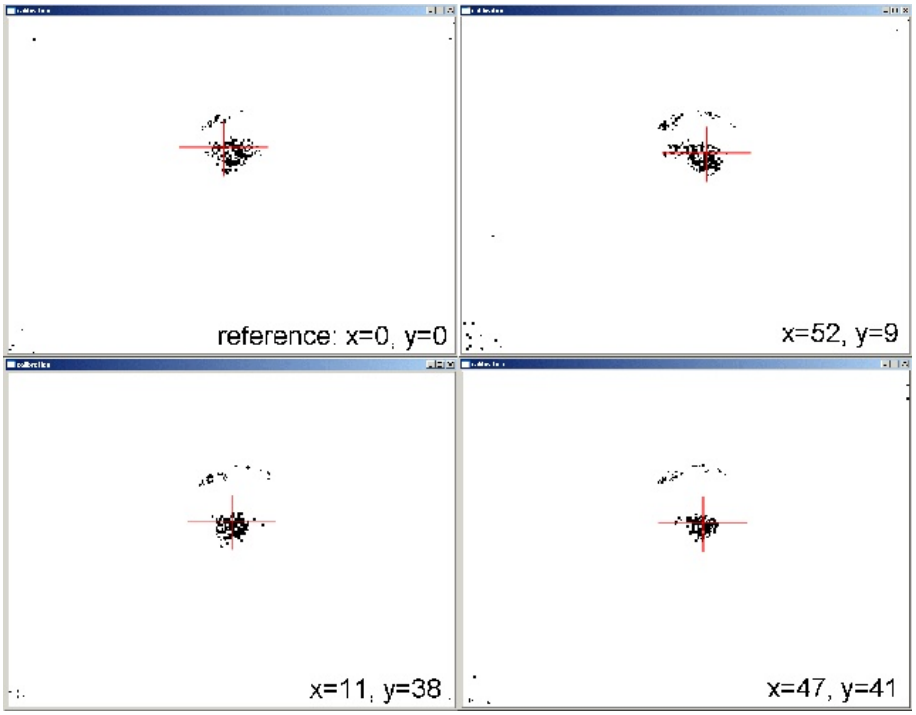

Figure 4.5: Locations of iris centers found for four corners of the rectangular scene

At each image, the location of the iris center is recorded. These four point form a rectangular basis to map any point back into the image coordinates. As long as the user does not change position with respect to the PC monitor,

and looks at a point inside the video frame, the center point of the iris will be inside the region formed by these four corner representations.

Continuous Eye Gaze Extraction

After calibration, the system is ready to watch the eye movements. Video frames start to be shown on the screen, and at the same time, the camera takes the images of the observer's eye corresponding to each frame. Every image of eye is then processed to find the iris center locations(see fig.4.5).

Having the calibration results at hand, the system calculates the point at which the eye is looking. This calculation is done assuming the eye is a sphere free to rotate around its center, and the screen to eye distance is too large than the radius of the eye sphere. Under these assumptions, the calculation becomes the similar triangles problem. For example, if the iris center is found at points P_L and P_R at the calibration stage. And it is now at point P_0, such that

$|P_0P_L|/|P_0P_R| = c$

then the point of interest on the screen can be found as;

$|I_0E_L|/|I_0E_R| = c$

where I_0 is the point of interest, E_L and E_R are left and right edges of the screen(see fig.4.6). This logic is only to find the horizontal location (x-component) of the point. The same calculations are to be performed to find the vertical component of the point.

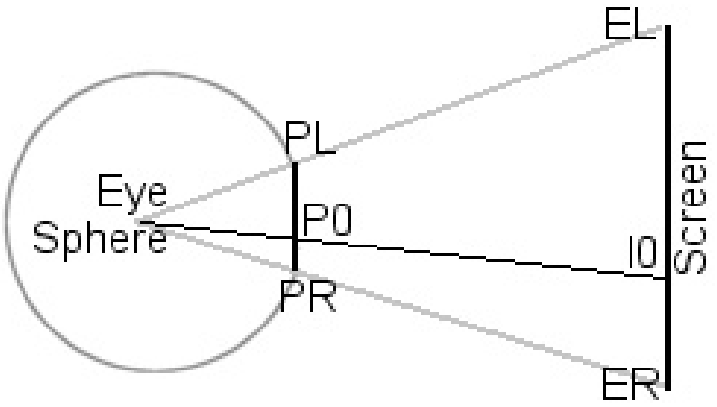

Figure 4.6: similar triangles in calculation

CHAPTER 5

SIMULATIONS AND PERFORMANCE ANALYSIS

In this chapter, the results of the algorithms used in both the scene observer and human eye interface parts of the system will be obtained for each individual system, and then a comparison about the similarity of the scene observer's results and the human-acquired data will be performed.

In this respect, the first section will include the information about the scene observer itself. Human eye interface data will then be investigated.

5.1 Simulation Results of Scene Observer

The scene observer is supposed to deal with a video stream of a scene, and therefore a video file is fed as an input. Throughout this section, some snapshots will be illustrated to give an idea about the overall test video.

The test video stream is a 20 sec. at 25fps color video with dimensions 320x240 pixels. In the scene, a stationary person holds two books in his hands. One of the books is flat black, and the other has yellow-red tones on its cover. The books are moved at different times and finally put together on the center of the scene. The actual video is provided in the CD attached at the back of this thesis.

Figure 5.1: A snapshot of motion and color channels where the black book is under motion and orange one is stationary

The scene observer processes the input stream according to the three user inputs entered at the beginning of each run. These inputs are those described in section 3.1.4, namely, color, motion, and habituation coefficients. Output of the scene observer (i.e. the coordinates of the point of focus) can be changed by modifying these parameters according to the current needs:

Qualitatively speaking, it can be stated that, adjusting color or motion parameters, interest is increased for colorful or moving objects. Low habituation values trigger rapid changes of the point of focus instantaneously, making an unstable focus point. As the value of the habituation parameter is increased, the movements of the focus slow down, and the focus gains more steadiness.

5.2 Human Characteristics

In order to observe the human eye gaze movements, an experimental setup is built using the human eye tracker introduced in section 4.1.2. The camera and lens are fixed on the bottom side and the ambient light is blocked inhibiting any entrance within the human eye tracker device. Without this ambient light shading, the built in automatic white balance system of the camera continuously changed the image brightness, causing the input to have an intolerable noise at each frame. After the very first experiments, it was further established that the white color of the aperture cone of the eye tracker causes similar problems, and the inner side of the cone was coated with a light-pink colored material which is found to be the closest color to an average face skin.

During the experiment, the same test video as shown to the scene observer is played on the PC monitor. The video window is resized to be 640x480 pixels. The remaining parts of the computer's screen are kept as bright as possible. The light coming from the non-video parts of the screen is used to illuminate the eye of the subject.

While the subject is watching the video stream, the camera takes pictures of eye continuously. However, it is noticed during the experiments that the computer used in this experimental setup (A 2100MHz PC with 512KB RAM on it) is not capable of processing every eye images in a short enough period of time to maintain the frame rate of 25 fps. Therefore, it is decided to perform the algorithms for every fourth image of eye. During the period in which other three frames are shown, necessary calculations as given in sections 3.1-4.1 are done, and the coordinates of the iris center which are recorded in a file for

Table 5.1: Calculated weight parameters for unbiased subjects

Subject	Color %	Motion %	Habituation %
Set 1	0	20	80
Set 2	50	30	20
Set 3	0	90	10
Set 4	0	100	0
Set 5	10	90	0
Set 6	10	90	0
Set 7	0	10	90
Set 8	0	20	80
Set 9	0	0	100
Set 10	0	100	0
Average	7	55	38

further use, are calculated.

5.2.1 Unbiased Subjects

In the first experiments, the subjects are not informed about the purpose of the experiment, and are only asked to watch a movie without looking outside of the video window. The data obtained from these experiments are then compared to that from the scene observer's results. At every frame, the distance between the human and computer pixels is computed and the Least Mean Square (LMS) distance is calculated for every possible weight parameter change. Then, found motion, color, and habituation parameters are assigned to corresponding experiment subject.

After three weight parameters are assigned to each set, the average set of these parameters are calculated to find which values of parameters can be used to represent a normal human. The average values of these parameters can be seen on table 5.1. Looking at that unbiased data we can conclude that a human

Table 5.2: Calculated weight parameters for biased subjects

Subject	Color %	Motion %	Habituation %
Set 1	0	100	0
Set 2	0	10	90
Set 3	0	100	0
Set 4	0	90	10
Set 5	10	30	60
Average	2	66	32

is more interested in motion than color features of the scene he is looking at. Habituation values also gets a considerable place. It is because of the fact that the human beings do not tend to make rapid, abrupt eye movements. As discussed before, higher habituation weights cause the eye movements to be smoother.

5.2.2 After Biasing

In order to see how much the weight parameters change with guidance, or with the given context, another set of experiments are performed.

In these second set of experiments, subjects are asked to look at any moving thing within the window, again without looking outside of the video window. The same LMS method is applied to find the corresponding weight parameters (see table 4.2 on pp.50).

As can be seen from the tables, the encouragement to look at the moving parts resulted in a 20% increase in the motion weight parameter.

5.2.3 Parameter Extraction with Neural Network: Computer learning to watch like a certain human

While trying to make a robot gaze in a more human-like manner, we incorporated a learning module based on artificial neural networks. Such a learning module enables the hardware (a robot, or here a PC) to gaze in a way to mimic a certain type of human (women, children, laymen from the city, villagers etc.). Learning is done by finding the weight parameters of each channel that was previously calculated using LMS. Here we determine the weight parameters of each channel by a layered neural network using the back propagation algorithm.

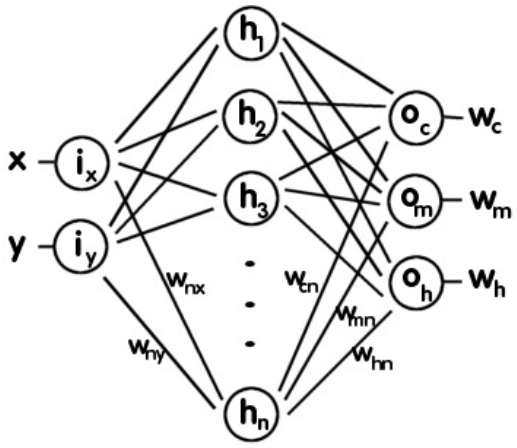

Figure 5.2: The topology of three layer neural network

The topology of the neural network should be such that it has two input neurons for the x and y coordinates found by tracking the eye, a hidden layer of n neurons, and the output layer having as much neurons as the number of channel-weights (previously called w_c,w_m, w_h).

For every frame, the network is fed with the coordinates of the gaze point. The resulting values are compared with the desired ones and the error between these

real and desired output patterns is tried to be minimized by backpropagating it through the neural network. This process is repeated until the change in error is negligible in the last turns.

While training an artificial neural network, it is also necessary to avoid overfitting or memorizing. To achieve this, one third of the available data, which is called 'evaluation set', is excluded from the training mechanism and is only used to test the network if it started memorizing the training set. After some point during the training, if the network is too much trained, it starts to memorize the training set data causing the errors for the evaluative data start to increase, while errors for the training set continues to decrease. If the training is not stopped at this point, the network would yield non-satisfactory results for actual data that is not used in the training set.

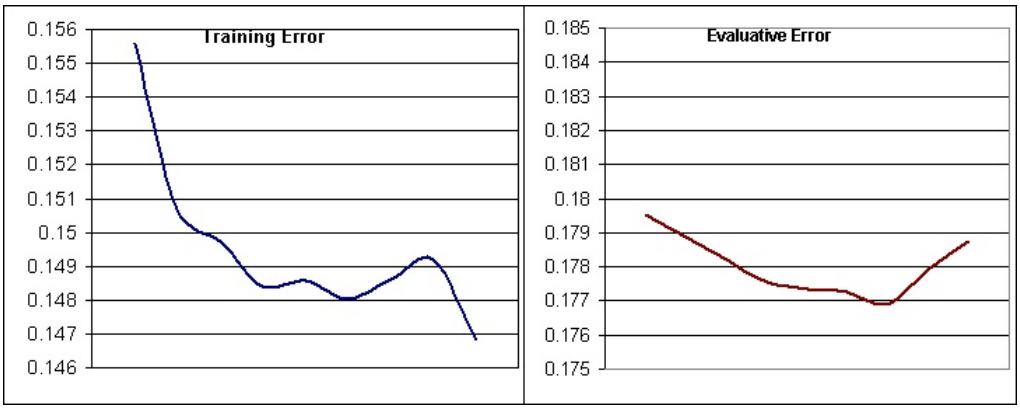

Figure 5.3: The error figure during the training process. Evaluation is performed to avoid memorizing

The results of the neural network based weight extraction and the LMS method are shown in table 4.3. It is seen that the results are slightly different. However, a trained network more likely generates weight parameters that are valid for a general class of human than the LMS method since an artificial neural network has a generalization capability. The results of the LMS method are quite

Table 5.3: Weight parameters by LMS and NN

Method	Color %	Motion %	Habituation %
LMS	7	55	38
NN	15	46	39

specific for this subject set and does not make any generalization.

The advantage of the neural training approach is that the system can be trained with a set of people with a certain property (age group, gender etc.) and the system will gaze (or behave) like that specific trainer group. If a non-group-member subject is compared with a member, the instantaneous errors will be higher for that outlying subject.

As an example, a child is taken as a subject and data is collected with the eye tracker. However, it is not included in the training set. The error data for the child with a group member is compared in fig.5.4. In the error graph, it is seen that most of the time the error for the child is higher than the adult. The only exception is that during the second quarter of the overall video. The reason for this is a very small movement in the colorful region of the scene. Adult subjects seem to have missed this little movement, however the child's attraction easily focused to the relevant area.

In a second example, the NN is trained using only the male subjects' data. Then, a male and a female subject (both been excluded from training set) is compared according to the error figures(see fig.5.5). It can be seen from the figure that the male subject's error is lower than that of the female's. This is because our system now mimics 'male humans' instead of a general group. Similarly, many other sub groups (farmers, students, women, soldiers etc) may be taught to the system if enough data can be obtained from relevant subjects.

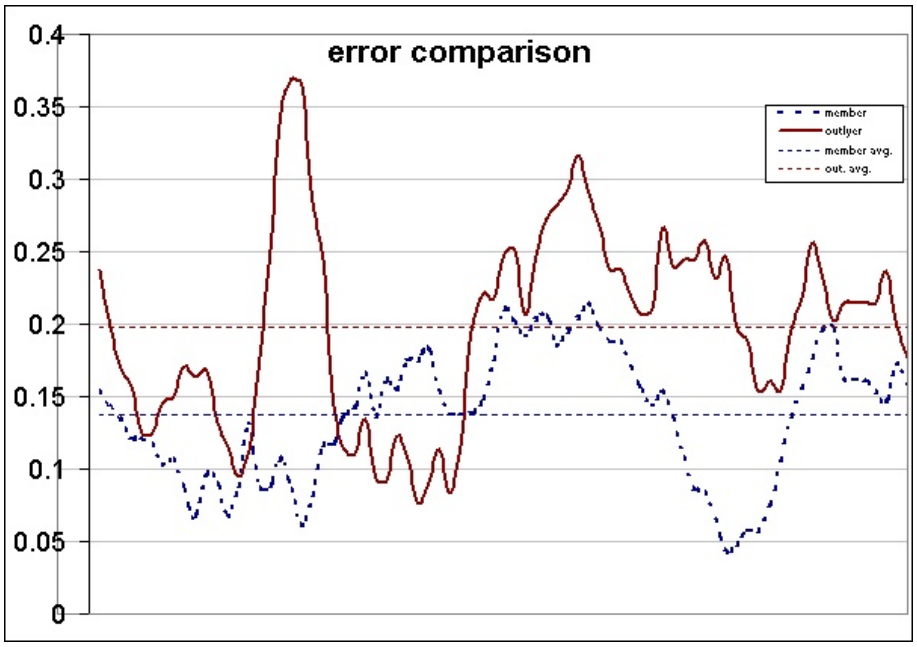

Figure 5.4: The error figure for the group member is lower than an outlyer(In this case a child of 10 yrs old)

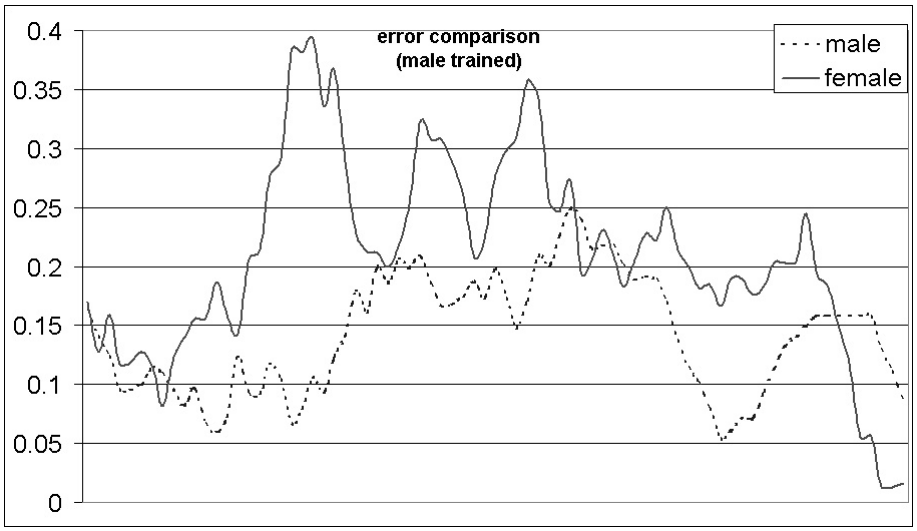

Figure 5.5: The error figure for two individuals (1 male,1 female) which are not included in training set when the network is trained with Male subjects.

CHAPTER 6

SUMMARY AND CONCLUSION

In this thesis, a multi channel visual system that can learn to gaze in a way similar to a human or a set of human belonging to a class with same characteristics like age, gender, profession, etc. is developed and its performance is tested with some experiments conducted with the actual human visual systems.

The developed system has three channels as input. For each of these channels, there is a preattentive feature (which are colorfulness, motion, and habituation in this implementation) assigned. Every channel is processed getting the main input image and applying relevant algorithms to obtain the feature map related to that specific channel. Some of these algorithms are colored eye correction for better performance in color eyed subjects, LLS algorithm to find the center of iris relying on the fact that the center of an ellipse lies at the center of the longest horizontal line passing through that ellipse. Although the human brain processes different tasks in parallel, our system does not have this capability. Instead, it depends on its high speed to process each channel one after another. During the experiments, a frame rate of 22 fps is achieved using a Celeron^{TM} 2.4GHz - 512KB RAM personal computer.

All of the channels are then linearly combined into one master channel by multiplying each channel with a weight parameter, and summing them up. The process then decides the most interesting point to look at, that is, the direction of attention is determined.

In order to compare the outputs of the computer vision system with that of the human, experiments are performed with human subjects. A weight parameter set of three entries is assigned to each subject in these experiments. This way, an average human's motion, color, and habituation parameters are obtained. As a second step of these experiments, the subjects are encouraged to look for the movement in the scene, and the resulting average movement-watching human characteristics have shown an expected increase (which is 20% in this case) in the parameter under bias.

Robot vision systems mainly rely on high resolution imaging technologies, and complicated image processing algorithms. And, with the increased complexity of the programs, the need for faster computer systems arises. However, a contribution can be made introducing the human like vision attributes. That is, the processing power can be directed more to where it is needed more, of course not fully neglecting the other parts but reduce their importance gradually.

The implemented system can be used for this purpose. It directs the focus of attention to where it is really needed, in a blink, just like the reflexes of human beings. More complex procedures are then able to spend their full power on the necessary parts of the scene image.

Although only three channels are used in the implementation, the system is flexible to accept as many different channels as wanted. For example, color channel may be separated into three channels of main colors, or, with addition of a second camera to the system, depth fields, or lustre channels can be added. Including more channels would probably improve the system performance since the system at this stage is not capable of handling 3-D features for example, which humans are experts at.

The system behaves like human visual system given the correct parameters found in the experiments. However, these parameters are just averages. As a further step, the system may be made mimic a person introducing a neural network based top-down control mechanism.

6.1 Future Work

Several researches based on this work may be conducted in the future. In order to have a more efficient generalization with the neural network based learning mechanism, the training set may be improved in size. The more subjects are taken into the training set, the better performance learning mechanism will give. As a second direction of researches, real parallel processing compatibility may be added. Parallel working multi CPU's, each is an expert of a specific feature, would achieve faster and better results.

As mentioned throughout the thesis, only features which can be extracted using a single camera are processed in our work. With the addition of a second camera, the remaining features like depth and lustre can be added as individual channels. Moreover, depending on the application more pattern channels may be added. This way the system would be able to detect smaller details in the scene which were totally unknown for current version. The whole system can be added as a module into other robotic systems where visual tasks are to be performed.

REFERENCES

[1] Rowel Atienza and Alexander Zelinsky. Active gaze tracking for human-robot interaction. In *International Conference on Machine Intelligence*, PA USA, 2002.

[2] Jean-Christophe Baccon, Laurence Hafemeister, and Philippe Gaussier. A context and task dependent visual attention system to control a mobile robot. In *Proceedings of the 2002 IEEE/RSJ, International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, October 2002.

[3] John Bell and Zhisheng Huang. Seeing is believing: A common sense theory of the adoption of perception-based beliefs. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, (13):133–140, 1999.

[4] B.K.P.Horn and B.G.Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[5] Patrick Cavanagh, John Boeglin, and Olga Elzner Favreau. *Perception of Motion in Equiluminous Kinematograms*, volume 14, chapter Perception, pages 151–162. 1985.

[6] C.Breazeal and B.Scassellati. A context-dependent attention system for a social robot. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI99)*, Stockholm, Sweden, 1999.

[7] Richard Dawkins. *The Blind Watchmaker*. TUBITAK Press, Ankara, 2 edition, 2002.

[8] Bruce A. Draper, Kyungim Baek, and Jeff Boody. Implementing the expert object recognition pathway. *Machine Vision and Applications*, 16(1):27–32, 2004.

[9] Nick Efford. *Digital Image Processing, A Practical Introduction Using Java*. Addison-Wesley, England, 2000.

[10] Ramesh Jain, Rangachar Kasturi, and Brian G.Schunck. *Machine Vision*. McGraw-Hill inc., Singapore, 1995.

[11] Kyung-Nam Kim and R.S.Ramakrishna. Vision-based eye-gaze tracking for human computer interface. (0-7803-5731-0/99), 1999.

[12] Jeffrey L.Krichmar and Gerald M.Edelman. Brain-based devices: Intelligent systems based on principles of the nervous system. In *Proceedings of the 2003 IEEE/RSJ, International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2003.

[13] Jeremy M.Wolfe. *Visual Attention*, pages 335–386. Academiz Press, San Diego, CA, 2 edition, 2000.

[14] Jeremy M.Wolfe, Nicole Klempen, and Kari Dahlen. Postattentive vision. *Journal of Experimental Psychology: Human Perception and Performance*, 26(2):693–716, 2000.

[15] Peter N.Prokopowicz, R.James Firby, Roger E.Kahn, and Michael J.Swain. Gargoyle: Context-sensitive active vision for mobile robots, January 1996.

[16] Richard O.Duda, Peter E.Hart, and David G.Stork. *Pattern Classification*. John Wiley-Sons, Inc., USA, 2 edition, 2001.

[17] Aude Oliva, Antonio Torralba, Monica S.Castelhano, and John M.Henderson. Top-down control of visual attention in object detection. *Journal of Vision*, 3(9):3, October 2003.

[18] William K. Pratt. *Digital Image Processing.* John Wiley and Sons, 2nd edition, 1991.

[19] Intel Research. Intel open source computer vision library, opencv, http//www.intel.com/technology/computing/opencv/index.htm, August 2005.

[20] Giulio Sandini, Paolo Questa, Danny Scheffer, Bart Dierickx, and Andrea Mannuci. A retina-like cmos sensor and its applications. In *Proceedings of 1st IEEE SAM Workshop*, Cambridge, USA, March 16-17 2000.

[21] Cagatay Soyer, Isil Bozma, and Yorgo Istefanopulos. A mobile robot with a biologically motivated vision system. In *Proceedings of IROS'96 IEEE/RSJ, International Conference on Intelligent Robots and Systems*, Osaka, Japan, November 1997.

[22] Cagatay Soyer, Isil Bozma, and Yorgo Istefanopulos. Apes: Actively perceiving robot. In *Proceedings of the 2002 IEEE/RSJ, International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland, October 2002.

[23] Kazuyuki Takahashi, Junya Tatsuno, and Hisato Kobayashi. Human like active vision for service robot teleoperation. In *Proceedings of the 1999 IEEE, International Workshop on Robots and Human Interaction*, Pisa, Italy, September 1999.

[24] A Murat Tekalp. *Digital Video Processing.* Prentice Hall PTR, 1995.

[25] Ales Ude, Christopher G. Atkeson, and Gordon Cheng. Combining peripheral and foveal humanoid vision to detect, pursue, recognize and act. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2003.