

Costa Rica Institute of Technology

Electronics Engineering School at the Costa Rica Institute of Technology
Computational Physics Laboratory Zernike Institute of Advance Materials
Faculty of Science and Engineering at University of Groningen



Design and Implementation of an Autofocusing Algorithm for the Goniometric Robotic Apparatus for Compound Eyes

A thesis submitted in fulfillment for the
Licentiate degree in Electronic Engineering

by

Mariana Guzmán-Fonseca

September 2017

INSTITUTO TECNOLÓGICO DE COSTA RICA

ESCUELA DE INGENIERÍA ELECTRÓNICA

PROYECTO DE GRADUACIÓN

ACTA DE APROBACIÓN

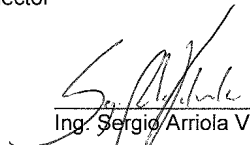
Defensa de Proyecto de Graduación
Requisito para optar por el título de Ingeniero en Electrónica
Grado Académico de Licenciatura
Instituto Tecnológico de Costa Rica

El Tribunal Evaluador aprueba la defensa del proyecto de graduación denominado **Design and Implementation of an Autofocusing Algorithm for the Goniometric Robotic Apparatus for Compound Eyes**, realizado por la señorita **Mariana Guzmán Fonseca** y, hace constar que cumple con las normas establecidas por la Escuela de Ingeniería Electrónica del Instituto Tecnológico de Costa Rica.

Miembros del Tribunal Evaluador


Ing. José Miguel Barboza Retana.MSc


Profesor lector


Ing. Sergio Arriola Valverde.Lic
Profesor asesor

Cartago, 22 de setiembre, 2017

Declaration of Authorship

I declare that the graduation project is entirely made by myself, with the help of scientists and engineers from the University of Groningen and the Costa Rica Institute of Technology, who gave me the opportunity to collaborate with the GRACE project located in the Netherlands. In the cases I have used literature, I mentioned sources by references. Therefore, I assume total responsibility for the graduation work done and for the content of the corresponding final report.

Signed: 

Date: 22-Setiembre-2017



Supervisors:

Ing. Sergio Arriola-Valverde.Lic
Costa Rica Institute of Technology

Dr.Ing.M.Muñoz-Arias
Costa Rica Institute of Technology

Prof.dr.D.G.Stavenga
University of Groningen

Assessment Committee:

Ing. José Miguel Barboza-Retana.MSc
Costa Rica Institute of Technology

Ing. Faustino Montes de Oca-Murillo.MSc
Costa Rica Institute of Technology

Abstract

The goniometer robotic apparatus for compound eye at computational physics laboratory at University of Groningen is requiring an autofocus algorithm that is able to capture on focus images for the analysis of compound eye insect for characterization. In the absence of an adequate algorithm that manages to put the camera in a right place, GRACE automation is difficult to search for respective image.

That is why, through the present thesis has been designed and implemented an algorithm of autofocus structured in modules which improve the organization of the process to achieve an adequate analysis of compound eyes.

It is proposed an algorithm that executes five techniques of autofocus capability to determine the sharpness in images in several channels, as well as to use some of that techniques to improve the positioning of the camera and acquire focused images.

Keywords: Algorithm, autofocus, GRACE, measure, vision.

Resumen

El dispositivo goniómetro robótico para ojos compuestos (GRACE) del laboratorio de la Universidad de Groningen está requiriendo de un algoritmo de autoenfoco que logre capturar imágenes enfocadas para el análisis de los ojos compuestos de insectos para su caracterización. Ante la falta de un adecuado algoritmo de autoenfoco que logre posicionar la cámara en un correcto lugar, se dificulta la automatización de GRACE para buscar tales imágenes.

Por eso mediante esta tesis se ha diseñado e implementado un algoritmo de autoenfoco estructurado en módulos los cuales mejoran la organización del proceso para alcanzar un adecuado análisis de ojos compuestos.

Se plantea un algoritmo que ejecute cinco técnicas de autoenfoco capaces de determinar la nitidez en imágenes en varios canales, así como utilizar alguna de esas técnicas para mejorar posicionar la cámara y adquirir imágenes enfocadas.

Palabras Claves: Algoritmo, autoenfoco, GRACE, medida, visión.

Dedicatory

A God, for giving me the strength to go ahead and finish this academic process for which I have given so much effort, for giving me perseverance patience and wisdom to get here and to the Virgin Mary for her guidance.

Dedicated to my mother forever to give me her unconditional love, her example as being a good professional, her support to keep going, motivating me to never overcome the difficulties and achieve my dreams.

To my sister, to inspire me to go ahead, tell me I can do it and always support me.

My grandmother, who although she is not physically, I leave many teachings of perseverance, dedication and never give up.

To my grandfather, to be always in the important moments of my life.

To my boyfriend, for giving me his support in the most difficult times, for all the hours waiting at the university to transport me to my house and his love.

Acknowledgements

There are people who want to thank you for your support during the development of this project. My sincerest appreciation a Dr.Ing.M.Muñoz-Arias for helping me to get the project abroad as well as giving me your guide to how to handle the project development, answering the questions and advising me.

To Prof.Dr.D.G. Stavenga for accepting me to collaborate in his research, I am very grateful for giving me the opportunity as well as his availability to answer questions.

The supervisor in Costa Rica, Ing. S. Arriola-Valverde.Lic for being a helpful person in the elaboration of the written report and giving me some ideas how solve the problem.

My assessment committee, Ing. J.M. Barboza-Retana.MSc and Ing.F. Montes de Oca-Murillo.MSc for being part of the project evaluation and its professional contributions.

I thank H. L. Leertouwer for his pleasant attitude and his willingness to provide the helping to take photographs that I required include in the project. Ketan Sahu both for sharing the implementations of this project to complement my designs as well as to help me bought some train tickets.

The John K. Douglass for answering some questions and take the time to explain some ways to identify flies and give me some other helpful information.

To the friends I have known at the Costa Rica Institute of Technology, Alberto Garín Campos for being a tutor and guide, a Daniel Viquez Gómez for his company and help during the studies and Alejandro Nevares Luis for his support, the three thank you for their friendship during the whole career. As well as several colleagues that I met in the university training process that were a great help to finish the career.

Contents

Declaration of Authorship	ii
Abstract	iv
Resumen	v
Dedicatory	vi
Acknowledgements	vii
List of Figures	x
List of Tables	xiii
Abbreviations	xiv
Symbols	xv
1 Introduction	1
1.1 Objective and Document Structure	4
2 Preliminaries	5
2.1 Image	5
2.1.1 Digital image processing	6
2.2 Compound vision	7
2.3 GRACE	9
2.3.1 System Operation	9
2.3.2 Visual mechanism	10
2.3.3 Mechatronic system	11
2.3.4 GRACE versions	12
2.4 Autofocus System	14
2.4.1 Microalgorithm	15
2.4.2 Macroalgorithm	15
3 Autofocus algorithm Implementation	17
3.1 Solution Analysis	17

3.1.1	Solution 1: Full custom autofocus algorithm	18
3.1.2	Solution 2: Autofocus algorithm using frameworks for image processing	19
3.1.3	Solution 3: Autofocus algorithm adaptation	19
3.2	Solution comparison and selection	20
3.3	Algorithm Design	22
3.3.1	Modules Description	23
3.4	Validation of Algorithms	26
4	Methods and Materials	28
4.1	Setting camera	28
4.2	Data Acquisition	30
4.3	Image Processing	33
4.3.1	Algorithms Explanation	33
4.3.1.1	Energy of Gradient	34
4.3.1.2	Energy of Laplacian Autofocus Algorithm	35
4.3.1.3	Gaussian Derivation Autofocus Algorithm	36
4.3.1.4	Tenengrad Autofocus Algorithm	37
4.3.1.5	Vollaths Correlation Autofocus Algorithm	39
4.4	Materials	43
5	Results	46
5.1	Setting camera results	46
5.2	Data acquisition results	48
5.3	Autofocus algorithm results	52
5.3.1	Normalized Algorithms	58
5.4	Data extraction results	63
6	Discussion	69
7	Conclusion and Recommendations	73
7.1	Conclusions	73
7.2	Recommendations (Outlook)	75
A	List Code	79
B	Features Camera	88
C	Hardware	91

List of Figures

1.1	GRACE machine,first version takes to [1]	2
2.1	Digital image representation (2.1), take it to [2]	6
2.2	Bee Apposition Compound Eye, adapted to [3], where the ommatidium is outlined in red box,green line is a rhabdom, black arrows meaning the light incidence and interommatidial angles formed between the black arrows.	8
2.3	General Control Diagram of the GRACE machine first version, took to [4]	9
2.4	Visual mechanism sketch of the photomicroscope used to photograph, took to [5]	11
2.5	Adaptation of the epi-illumination microscope which generates an image of the insect eye through the lens and the mirror used to acquire image data. The letter L represents lenses, D diaphragms and LM is emi-reflecting mirror, take to [6].	12
2.6	Illustrative description of the axes distribution in movement system of the GRACE first version, adopted to [4]	13
2.7	GRACE second version that represents a new adaptation goniometer, the goniometer has the next parts as shows in the figure where 1) it is the microscope located in the camera axis, 2) light source, 3) XYZ mechatronic system and 4) the visual mechanism which is under construction.	13
2.8	Flow diagram which shows the autofocus process of how catch an image, adapted to [4].	14
2.9	Macroalgorithm Interface Schematic.	16
3.1	High-level module solution of autofocusing algorithm design, own authorship	22
4.1	Camera configuration process diagram, own authorship.	29
4.2	Data Collection process diagram, own authorship.	30
4.3	Data acquisition system showing the two blocks that comprise them, computer data reception shown in the left block and the camera positioning circuit represented by the other blocks to the right.	31
4.4	The Data flow of start communication with Arduino in automatic condition, own authorship.	31
4.5	Front view photo of the <i>Sarcophaga</i> fly which is appreciated the head and the compound eyes, photograph taken by H.L.Leertouwer.	44
4.6	Front view photo of the <i>Scathophaga</i> fly which is appreciated the head and the compound eyes,photograph taken by H.L.Leertouwer.	45
5.1	Format showing the live video that is observed in the computer screen, representing the full size compound eye image without the desired region selected, took to [7].	46

5.2	The image zoom format of the insect eye is illustrated on a specific area. The size corresponds to 200x200 pixels, own authorship acquired from Matlab software.	47
5.3	Block diagram of the AutoFocusGRACE Function part of the Data Collection automation program. Four different colours, the red arrows represent the path taken by the data flow, the blue box represents input parameters with the function needed to run the script, the green box represents the cycle that repeats until the number of Cycles are defined by the user and the black boxes are the commands to execute the algorithm, own authorship.	49
5.4	Block diagram of the AutoSlope Function part of the Data Collection automation program. Four different colours, the red arrows represent the path taken by the data flow, the blue box represents input parameters with function needs to run the script, the green box represent the cycle that repeats until the condition is reached and the black boxes are the commands to execute, own authorship.	50
5.5	Focus Measure values graphics represented in red, green and blue channels for the five autofocus techniques with <i>Sarcophaga sp</i> when scanning non stop in the best image.	53
5.6	Focus Measure values graphics represented in red, green and blue channels for the five autofocus techniques with <i>Scathophaga stercoraria sp</i> when scanning non stop in the best image.	54
5.7	Focus Measure values graphics represented in red, green and blue channels for the five autofocus techniques with <i>Calliphora sp</i> when scanning non stop in the best image.	55
5.8	Focus Measure Normalized values graphics represented in red, green and blue channels for the five autofocus techniques with <i>Sarcophaga sp</i> when scanning non stop in the best image.	59
5.9	Focus Measure Normalized values graphics represented in red, green and blue channels for the five autofocus techniques with <i>Scathophaga stercoraria sp</i> when scanning non stop in the best image.	60
5.10	Focus Measure Normalized values graphics represented in red, green and blue channels for the five autofocus techniques with <i>Calliphora sp</i> when scanning non stop in the best image.	61
5.11	Edge Detector 1 with a Threshold $t=0.2$, image obtained with the <i>Scathophaga stercoraria sp</i> with the Tenegrad autofocus technique in red,green,blue and monocromatic channels.	66
5.12	Edge Detector 2 with a Threshold $T=0.4$ with the Canny filter and Sobel filter without threshold, image obtained with the <i>Scathophaga stercoraria sp</i> with the Tenegrad autofocus technique in red,green,blue and monocromatic channels.	67
5.13	Edge Detector 3 with the Subpixel Edge detector Filter, image obtained with the <i>Scathophaga stercoraria sp</i> with the Tenegrad autofocus technique in red, green, blue and monocromatic channels.	68
B.1	Texto para la leyenda	89
B.2	Blacky USB3 Vision Camera Specifications	90
B.3	Blacky USB3 Vision camera General features	90

C.1 External circuit part of the hardware movement control	92
--	----

List of Tables

2.1	Movements System Classification	12
3.1	Minimum Requirements and Constraints	18
3.2	Solutions Comparison	20
4.1	Main hardware materials list for the GRACE goniometer	43
5.1	Summary of the maximum FM values obtained by each algorithm for autofocusing with <i>Sarcophaga sp.</i>	56
5.2	Summary of the maximum FM values obtained by each algorithm for autofocusing with <i>Scathophaga stercoralia sp.</i>	56
5.3	Summary of the maximum FM values obtained by each algorithm for autofocusing with <i>Calliphora sp.</i>	56
5.4	Differences between FM magnitudes are shown in percentages to note the difference between channels in <i>Sarcophaga.</i>	57
5.5	Differences between FM magnitudes are shown in percentages to note the difference between channels in <i>Scathophaga stercoraria.</i>	57
5.6	Differences between FM magnitudes are shown in percentages to note the difference between channels in <i>Calliphora.</i>	58
5.7	Out of focus and focus images resulting from autofocusing techniques of Energy Gradient, Tenengrad (the two best techniques) and Gaussian Derivation (the technique with most unfavorable results) with a 200x200 zoom of the <i>Scathophaga stercoraria sp</i> eye.	63
5.8	Out of focus and focus images resulting from autofocusing techniques of Energy Gradient, Tenengrad (the two best techniques) and Gaussian Derivation (the technique with most unfavorable results) with a 200x200 zoom of the <i>Sarcophaga</i> eye.	64
5.9	Out of focus and focus images resulting from autofocusing techniques of Energy Gradient, Tenengrad (the two best techniques) and Gaussian Derivation (the technique with most unfavorable results) with a 200x200 zoom of the <i>Calliphora</i> eye.	65

Abbreviations

CPP	C orneal P seudo P upil
D	D iaphragms
DOF	D egree O f F reedom
DPP	D eep P seudo P upil
EA	E levation and A zimuth A xes
EneGrad	E nergy of G radient
EneLap	E nergy of L aplacian
FACET	F ast A pparatus for C ompound E ye T opology studies
FM	F ocus M easure
GausDer	G aussian D erivation
GRACE	G oniometric R obotic A pparatus for C ompound E ye
ITCR	I nstituto T ecnol g ico de C osta R ica
L	L enses
LED	L ight E mitting D iode
LM	L ense emi-reflecting M irror
PP	P seudo P upil
RuG	R ijksuniversiteit G roningen
USD	U nited S tate D ollar
XYZ	X axis Y axis Z axis

Symbols

∂	partial derivation
∇	gradient
f	function of the image
g	light intensity

Chapter 1

Introduction

Nature shows an awesome life cycle which has been the subject of research to explain, to know and to deepen in areas that lack information in the scientific and technology community. An important field of survey has been vision systems, an area which its analysis has been fascinating.

A striking vision system in nature is found on arthropods, a kind of insects, and from which research remains such an attractive subject for the discovery that so far has been made on the optical properties of arthropods.

Arthropods have a compound vision, a concept that has been studied to understand the importance of its characteristics. Compound vision on insects is a structure which has a compound eyes, there are thousands of smaller eyes called *ommatidia*, where each ommatidia is a facet of the compound eye and it works as an independent optimal system.[8]

The *ommatidia* is a sensorial unit formed by photoreceptor cells capable of distinguishing between the presence and lack of light [9]. It is composed on the outside of a facet lens, conducting the light towards its interior where it finds a crystalline cone that is channeled in a *rhabdom*, optical waveguide of the ommatidium [3]. The *rhabdom* is a structure that receives the light coming from the *ommatidia* and the light is transformed into electrical signal (nervous impulse).

Each ommatidia has a cylindrical shape which generates an image from adjacent fields with a range around 270 or 360 degrees of vision that form a vision called a mosaic vision,

in other words, due to being formed from different light intensities which come from multiple directions and together with other environmental properties is how the image that perceives the compound eye is apposition (called apposition compound vision) [8].

The well developed visual system is found in butterflies and flies which have compound eyes and are considered to be highly visual animals. The characterization of compound vision has been done manually with conventional microscopes that do not have the appropriated movements producing a process that has been time-consuming.

Taking into account properties such as the way the light is reflected, the way the image is formed and the fast interpretation of the medium in the compound eyes, operations want to be replicated in modern artificial vision systems like an artificial navigation or to optimize optical systems as cameras, televisions, among others.

The Department of Computational Physics of The University of Groningen has developed many studies based on the eyes of the butterflies leading to a long process which has proposed to create a specialized device known as Goniometric Robotic Apparatus for Compound Eyes (GRACE)[1].

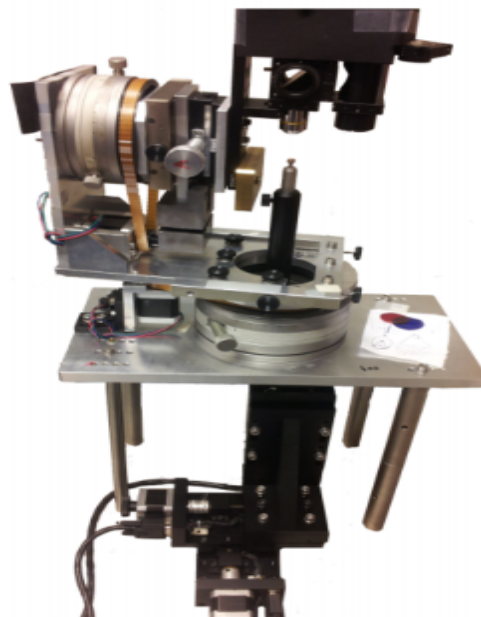


FIGURE 1.1: GRACE machine, first version takes to [1]

GRACE is depicted in Figure 1.1, the goniometer is a robotic scanner with six degrees of freedom (*DOF*), it means, has six movement axis. This goniometer has been used to characterize the vision space of butterfly eyes [10]. In general terms, this machine is

compound by two part that work synchronously. The first part is the positioning system where stepper motors are used for all mechanical movements which consider both XYZ-direction and two rotations known as azimuth and elevation axis. Recently, this part has been improved by being automated by the use of freely accessible hardware and software [1].

The second part is the visual system, which include cameras, lens and microscope with a good optical system. Visual system is taken over by autofocus and autocenter algorithms to capture the image. An important aspect is to get focused images so that we have searched for an autofocus algorithm that satisfies the work of images suitable for an appropriate analysis [11].

Also, one interesting algorithm which seeks correlating data in overlapping images of a butterfly eye taken by GRACE, has been done by Vargas [4], his algorithms are known as stitching algorithm which helps to find where is the greatest focal area concentration of the compound eye.

It has been found that such a specialized mechatronic device for compound eye characterization as GRACE is not the only existing device, a research and implementation by the biophysicist J. Douglass created a highly automated goniometer instrument called FACET (Fast Apparatus for Compound Eye Topography Studies), has been created to facilitate a quick mapping of compound eye parameters for investigating in visual areas [12].

On context, the visual part specifically the autofocus algorithms should be necessary to improve because it is not being validate, in other words, the autofocus algorithms have been adopted to achieve a focus measure of the image which has the compound eye but there is not a explanation why this algorithms work to determine if the image is in or out of focus. A student of Industrial Engineering who collaborated with GRACE project, has proven around twenty six possible algorithms which have been considered in the goniometric microscope but after several tests carried out five resulted appropriate [11].

Besides, one important thing that involves the autofocus is determine a good camera position which must be in focus to obtain a good quality image. So far both the autofocus and the camera position must be the main purpose in the project in order to help get a

suitable images analysis as well as contribute to improve the visual part of the GRACE machine to obtain compound eyes images.

1.1 Objective and Document Structure

The main objective of the project is the implementation of an autofocus algorithm, which is to add some modifications and adjustment to the existing GRACE autofocus algorithms, in order to improve and make sure the focus that processes images to search for the sharpest image within the set of images captured by the device. To that end, the designs of the autofocus algorithms are validated through revisions of the mathematical process giving interpretation to each one of the five algorithms most used, for this it is necessary several tests done online and offline the GRACE machine, tests prove that the indicated image which is the most focused (either a set of images or the live video captured) is the same image that the autofocus algorithms designate. Once acquired this analysis, understanding well the function and the results achieved by each algorithm, it is tried to design and to implement improvements to the mentioned algorithms.

This document has a structure by five chapters, which is detailed below: Chapter 2 it is a the preliminaries which contain theoretical concepts that are explained for a better understanding of the development of the project. Chapter 3 presents the proposed solution for the development of algorithm validation as well as the possible improvements that are required for the focus algorithms. In Chapter 4 present the methods and materials, here you can be able to found the explanation of each procedure block which is the way has been carried out for the implementation and together with the materials description. Chapter 5 shows the results obtained, Chapter 6 gives an analysis of the interpretation of the data. Finally, conclusions and possible recommendations for future research are presented in Chapter 7.

Chapter 2

Preliminaries

The chapter deals with concepts of relevant interest for the understanding of this project, in order to familiarize the reader with terminology of necessary issues that are addressed in the development of the solution. Recommended references [1-5,10,12-15,17].

2.1 Image

The image concept is explained as a bidimensional function $f(x, y)$, which is represented depending on the context, since this function $f(x, y)$ includes component such as illumination $i(x, y)$ and reflection $r(x, y)$ in the coordinate space (x, y) to represent a scene, [13] where the product of both functions provide the two-dimensional function as shown by:

$$f(x, y) = i(x, y)r(x, y) \quad (2.1)$$

The function value (2.1) depends on the amount of light falling on the scene (lighting) and reflection to occur on the object of interest, for example, the Figure 2.1 is depicted a $f(x, y)$. The illumination originates from the light source and the reflection depend on the characteristics of the object in the scene.

The image function regards to the bidimensional representation that exemplifies the quantization or sampling of analog images for processing and analysis in a computer, in

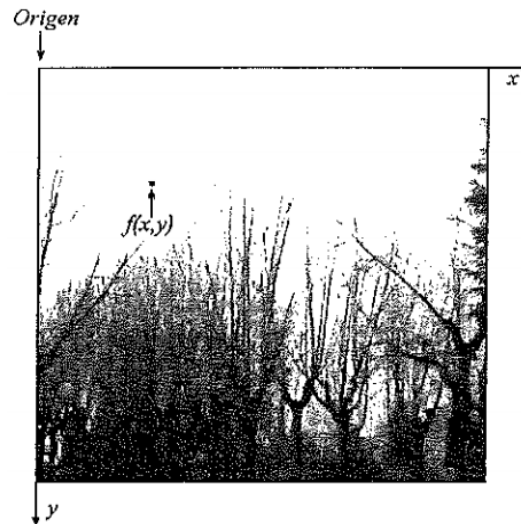


FIGURE 2.1: Digital image representation (2.1), take it to [2]

other words, it is the process of digitizing the image while retaining all the information of the analogue image [14]. The image is made up of a finite number of elements which have a specified value and location.

An important concept to emphasize in digital images to determine a sampling interval, is the frequency unit that is defined as the inverse of the distance unit in the plane of the image. In image processing, the distance unit is the pixel.

The pixel is the smallest basic unit of a scanned image on screen that can be manipulated by a computer when creating graphics, is composed of color or grayscale points, the pixels can be manipulated to different image procedures [15].

2.1.1 Digital image processing

The digital image processing is a way of transforming images into a modified one, in which both the input and the output of the process are images[16]. Through the procedure digital images are analyzed to extract, identify or detect information of interest that is in the images. Processing techniques are capable of enhancing the visual appearance of images to be viewed by an observer or to be suitably arranged to be perceived by machines. Also, image processing is considered a computerized process which can be branched into three groups such as low, midst and high level processes.

Low-level process involves basics functions such as contrast enhancement, image sharpening or reduce noise in images, it process has images both in its input and outputs.

Mid-level process works with segmentation functions (separate image into regions or objects), classification or recognition of individual objects. It process has images in its inputs but its output are elements extracted from images. The last one, the high-level process performs the perceptive functions normally associated with vision, as in image analysis of the recognized objects [16].

The above is a way to classify the processing, to great features, since there is a great amount of transformations or operations that can be performed on the images with the purpose of enhancing and improving them.

2.2 Compound vision

The study of composite eye structures is of great relevance due to the characteristics and properties of the compound vision, especially the study of the vision of butterflies and flies that are the main purpose of the project research known as GRACE. The eyes are a sensory organ that forms images from the incident light that comes from the environment [11]. In insects, the eyes can be classified into two groups, simple eyes and compound eyes. Simple eyes are organs with basic structures of vision that do not create true images of objects, lacking certain features such as focusing mechanisms. The human vision is like simple eyes.

Compound eyes are more complex structures because they are sensory organs capable of generating in the insect brain images that reflect the shape and color of objects [17]. Visual organ is formed, depending on the species, with 12 to 6000 or more receptive units called ommatidias which conform the compound eyes.

Each *ommatidia* is considered an independent visual unit, forming small images come together and set up the complete image. The *ommatidia* contains internal structures, which are shown in the Figure 2.2, like corneal cone lens (cc), facet lens (f), accessory screening pigments (sp), photosensitive *rhabdom* (rh, Green, it is to photoreceptor that distinguishes whether or not there is light), photoreceptor cells (rc), and axons (ax) [4]. It is emphasized that the separation between each ommatidia is formed interommatidial angles (ϕ) and so that the light received in each visual unit is limited by the acceptance angle (aa).

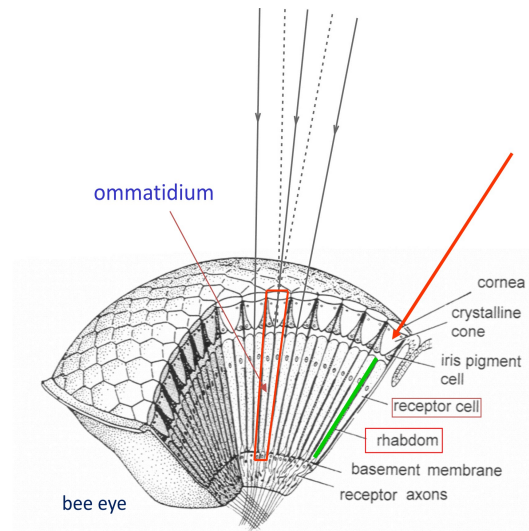


FIGURE 2.2: Bee Apposition Compound Eye, adapted to [3], where the ommatidium is outlined in red box, green line is a rhabdom, black arrows meaning the light incidence and interommatidial angles formed between the black arrows.

Besides, compound eyes can be classified into apposition and superposition eyes. In the apposition eye each *ommatidia* has dark pigments to prevent light passing between ommatidium which work separately from each other [8]. It is not very sensitive to light due to the small size of the pupil *ommatidia* where the light enters, but it is very accurate and it can be found in diurnal insects. Superposition eyes, on the other hand, are more sensitive to light but less accurate. This kind of eyes exist in nocturnal insects, where there are lighter pigments between the visual unit, a difference of the apposition eye[3].

On the one hand, the butterfly has apposition eyes and on the other it includes two characteristic of superposition eye. The reason why the butterfly comes from a larva, the larva is a nocturnal animal with superposition eyes but when it evolves to a butterfly it acquires the apposition eye structure.

When the analysis of the eye with the butterfly is done, a beam of light is placed on the eye, producing a reflection of the brightness of the color which is called eyeshine. The light enters the facet, continues through the corneal cone reaches the rhabdom and finally reaches the photoreceptors.

The light is reflected in the tapetum (it is a tissue that acts as a mirror that reflects light rays by increasing the available light) and a part of the light leaves the eyes. Eye brightness may be colored, which is produced by the reflectance spectrum of the tapetal mirror and the transmittance spectrum of the photoreceptor pigments if they are near

the rhabdom who guided the light. In butterflies this eye eyeshine, in most species, is red because of the presence of photoreceptors sensitive to red. [5]

2.3 GRACE

Its name is Goniometer Robotic Apparatus for Compound Eyes (GRACE) is a specialized 3D robotic scanner for capturing photographs in order to study and characterize the structure of the insect compound eye. GRACE, is based on the design of Stravenga, [5], has six stepper motors (which give six degrees of freedom, DOF), a goniometer, a digital camera, a LED light source and a four lineal joint (one of them moves the camera and the other three are to move in Cartesian coordinates axis). The system could be set in more detail below in general aspects.

2.3.1 System Operation

The operation of GRACE is through a system which directs the process, the general diagram is displayed below.

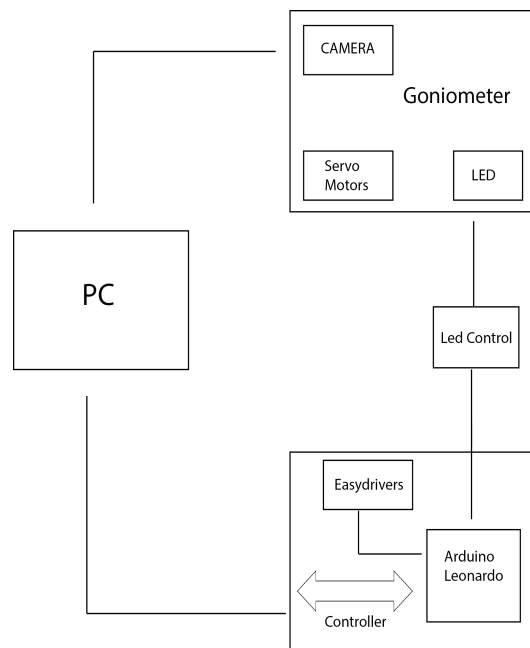


FIGURE 2.3: General Control Diagram of the GRACE machine first version, took to [4]

The system operation of GRACE Figure 2.3 is carried out by means of the three blocks union which are the computer, controllers and the goniometer. The computer has the

function of controlling the GRACE system using Matlab software. In addition, the Leonardo Arduino microcontroller is used which controls the six step motors “Easy-Driver” and the LED light source that are connected by serial port to the computer. Finally, the goniometer contains the camera which communicates via USB 3.0 directly to Matlab. The set of all makes possible the analysis of composite vision when photographing the eyes with a high magnification about $20\mu\text{m}$ which is projected in the image to be acquired.

2.3.2 Visual mechanism

The visual apparatus is composed by the microscope and a camera, the microscope has a optical part which is based on the design made by Stravenga. The camera adapted on GRACE is Point Grey Flea 3 USB 3.0 which is able of capturing image at up to 3.2 megapixel color image at 60 FPS getting a resolution of 1920 by 1200 pixels.

The Point Grey cameras are easy to integrate with a computer and the technology that uses is a variety of *CMOS* (Complementary Metal-Oxide-Semiconductor) which is a logic family of integrated circuits with [18] image sensors in ultra-compact way. Its camera has an interface designed to provide a variety of applications such as optical character recognition, gesture and object tracking among others, the camera features and more details are in the Appendix B.

It is important to mention that the system’s optical setup has been changed due to the incorporation of the camera. The light source and the camera are in different axis position because when they are directed through the mirrors that make up the microscope they interact separately as explained below. In figure 2.4 GRACE’s optical system is depicted.

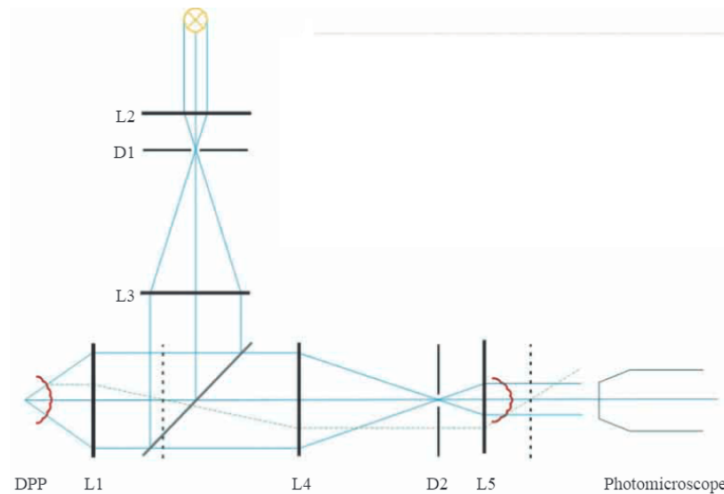


FIGURE 2.4: Visual mechanism sketch of the photomicroscope used to photograph, took to [5]

The optical part starts with a lens L1 has a large aperture where a light source is focused at lens L2 in its back focal plane and the field diaphragm D1 is positioned. D1 is in the focal plane of lens L3, which is confocal with L1 because of a half-mirror placed at 45 with respect to the optical axis of L1 and L3. A more or less parallel beam, depending on the size of D1, enters L1 and is focused on the deep pseudopupil (DPP) in the center of the butterfly's eye. The telescope lens pair L1 and L4 images the DPP in the back focal plane of L4, where diaphragm D2 is positioned. The image of the corneal pseudopupil (CPP), projected by lens L5, confocal with L4, is photographed by the Point Grey camera. The dotted lines are the back-focal planes of L1 and L5 [5]. The epi-illumination microscope has been modified as is depicted in Figure 2.5, in which the design is very similar to the previous one.

A new diaphragm is added to the epi-illumination system but retaining the same functionality. New part is included because it is required reflections to obtain the data due to a new semi-reflecting mirror where the behavior of light reflections and refraction are the same as in the Figure 2.4.

2.3.3 Mechatronic system

The mechanics of system movement have a high complexity because of their six degrees of freedom (six-DOF). Each axis of movement has a Wantai stepper motor that requires the current source of 2.5A and has 1.8 degrees angular step. Because the machine needs

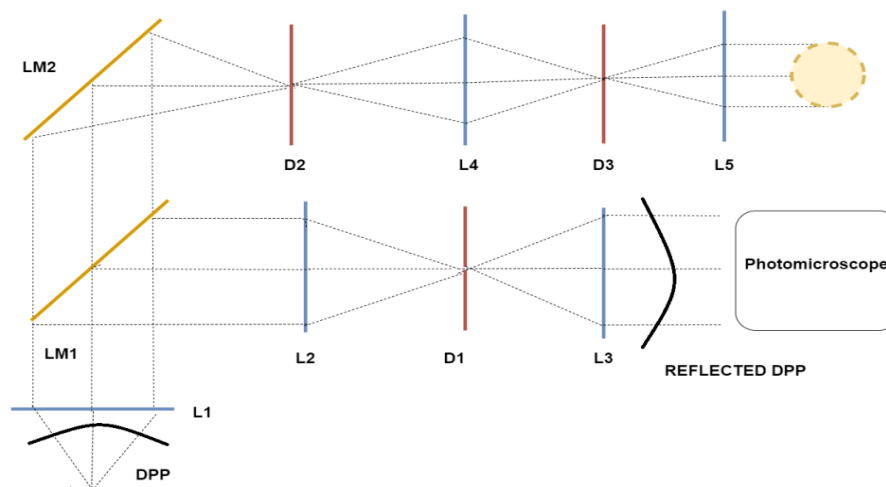


FIGURE 2.5: Adaptation of the epi-illumination microscope which generates an image of the insect eye through the lens and the mirror used to acquire image data. The letter L represents lenses, D diaphragms and LM is emi-reflecting mirror, take to [6].

precise movement, five motors have been configured to have an angular step smaller than the default engine set-up. The axes of the system are X , Y , Z , its two rotational directions (elevation and azimuth, EA) and camera, which are classified into three sets which can be seen in the Table 2.1.

TABLE 2.1: Movements System Classification

Stage	Description
XYZ Set	Three axis where the target moves
EA Set	The optical system move in this two rotational axis
Camera Axis	Linear axis where the camera moves around its visual joint

The separation is performed to explain the way the motion in the goniometer is given. The Figure 2.6 shows the distribution of the axes. With the elevation axis position it is determined by θ (it is only deployed in the range of 180 degrees) and the azimuth axis defined by ϕ (which moves in the range of 360 degrees). The set EA is the one that tracks the goniometric position.

2.3.4 GRACE versions

Necessary to improve the machine by making adjustment to achieve its automation to execute a good movements control and images analysis, details that have been corrected in the current GRACE. Even though, the later mentioned machine has worked properly, a new version of GRACE has been built to complement the researches.

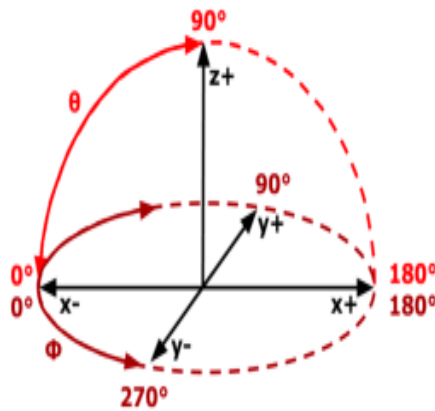


FIGURE 2.6: Illustrative description of the axes distribution in movement system of the GRACE first version, adopted to [4]

GRACE second version works as the same as the other, it means, the new scanner has two part the visual mechanism and the mechatronic system but it has different aspects which are described later. The camera position is the azimuth and elevation axes and also it is independent of the XYZ system. The camera is able to move in one axe with a DC motor instead of the steppers which in the first version let the camera move. Causing the target insect to move toward the camera instead of the camera moving around looking for the insect. The version is showed in the Figure 2.7, and it is worked on it because is not finished yet.

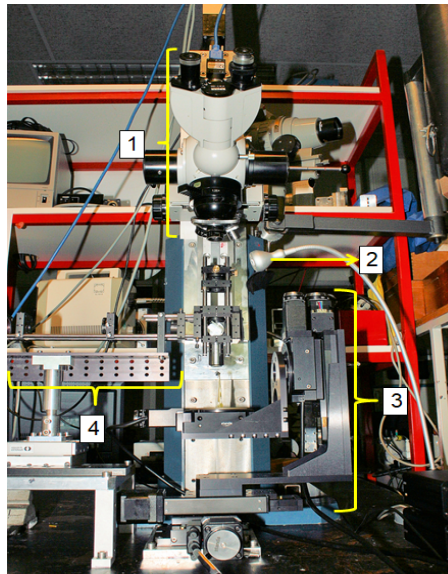


FIGURE 2.7: GRACE second version that represents a new adaptation goniometer, the goniometer has the next parts as shows in the figure where 1) it is the microscope located in the camera axis, 2) light source, 3) XYZ mechatronic system and 4) the visual mechanism which is under construction.

2.4 Autofocus System

The autofocus system is located within the visual system which plays a important role in the image processing, that helps to make spheroid surface maps or different sizes of facet maps of the compound eyes local curve. Autofocus algorithm is a automatic procedure that involves the microscope, the light source and the camera to work together with an autocenter algorithm in order to get images, in the Figure 2.8 shows the procedure to catch an image. Also, it is important to mention some assumptions, the camera must be perpendicular and the procedure should be done along the camera axis to obtain a good quality image [11].

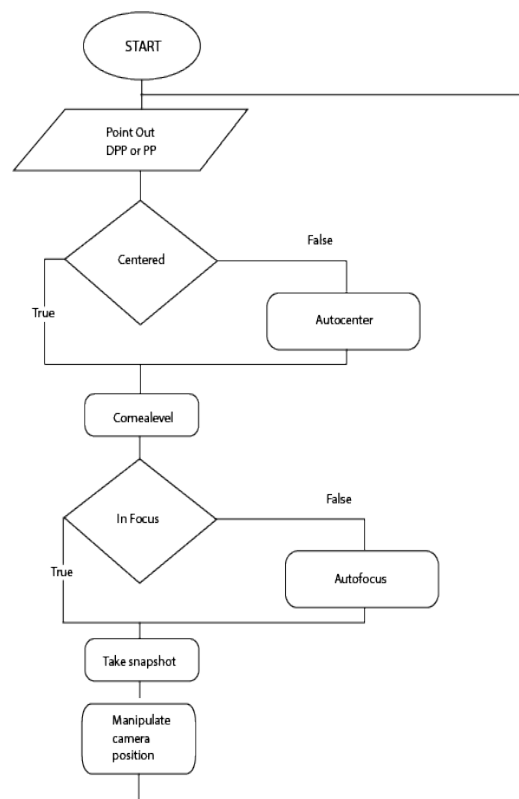


FIGURE 2.8: Flow diagram which shows the autofocus process of how catch an image, adapted to [4].

The Figure 2.8 serve as the explanation of the catch procedure where the conditions to acquire the image are verified, this entails a verification if it is in the deep pseudopupil (DPP) or in the pseudopupil (PP), then it is verified if the image that is being seen is centered, if it is true it continues, otherwise it is centered with a function called *Autocenter* the condition is performed if the image is centered. The condition of the image is centered is executed, the process is continued if the condition is true but

the *Autofocus* function is called when the condition is false, the snapshot is taken, the camera position is adjusted and the process is performed once more to check the process.

2.4.1 Microalgorithm

The microalgorithm refers to the autofocus algorithms for the vision system of the GRACE second version goniometer, involving the set of techniques that have been implemented to find the focus measure (FM), that means, the microalgorithm is the software programs that seek the highest value of FM which exhibit the high level of focus.

The focus measurements implicate mathematical procedures that have been used to evaluate the levels of sharpness in the images by mean to extract some characteristics present in the original images or to eliminate attributes that prevent extracting what is sought from the image [11].

The autofocus techniques are implemented by the following algorithms which have been chosen and these ones are:

- Energy of Gradient [19]
- Energy of Laplacian [16]
- Gaussian Derivation [20]
- Tenengrad [21]
- Vollath's Correlation [22]

2.4.2 Macroalgorithm

A correct positioning of the camera that captures the compound eye must be carried out using equipment adapted to GRACE version two, this implies a hardware interface that involves moving the camera to get images suitable for the desired analysis. The hardware interface working in conjunction with what the autofocus algorithms involve (known as the microalgorithm) is what is called the macroalgorithm.

The hardware interface has required several modules that interact to achieve the camera, these modules are shown in Figure 2.9.

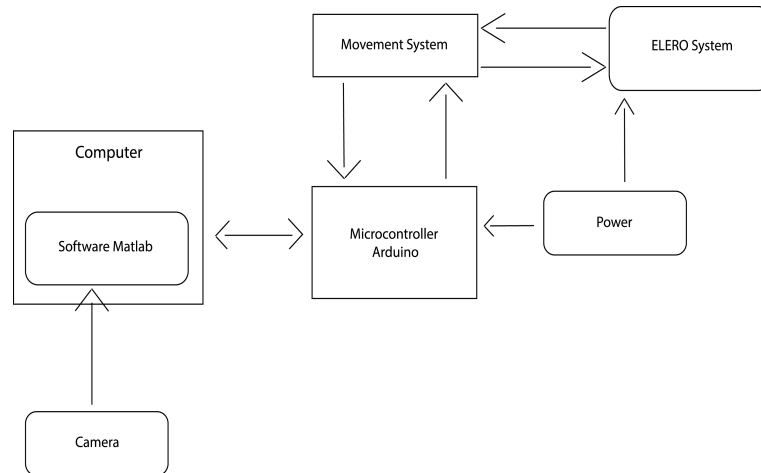


FIGURE 2.9: Macroalgorithm Interface Schematic.

It has been made use of three blocks which are a controller, a movement system and the Elero system. The microcontroller allows the control that interact between the computer and the positioning of the camera, is in charge of the instructions that are executed to move the camera through the signals that are sent by the computer. The motion system allows the direction of rotation of the motor depending on the instruction given to it, it refers to whether the camera moves up or down. While the ELERO system contains a DC motor and an ENCODER, system which holds and is the apparatus where the camera movement is given.

Therefore, the desired controller must be an Arduino mega that is being used to control the other motors and also extract data from the Elero encoder and the movement system which help to move up or down. After that, Arduino software exports data to a two USB connections towards the computer, one is a USB 3.0 with the camera and the other is using a serial communication. The complete circuit connections is found in the Appendix C (Figure C.1).

Chapter 3

Autofocus algorithm

Implementation

Chapter considers the development of the design process, validation of autofocus algorithms and implementation proposal to improve the focus level of the acquisition images through the GRACE instrument. To start in this process it is important to specify the requirements and constraints that must be fulfilled for the project design and its objectives. For this the Table [3.1](#) shows these details.

3.1 Solution Analysis

The current system of the GRACE goniometer is a prototype specialized in the study of composite eyes, the goniometer is built from the specific characteristics and necessities for the investigation, reason why it is not a commercial system, and for its improvement in the focus analysis, it is required to delimit the way to solve the project.

Three ways of solving the autofocus algorithm are presented, capable of giving an optimum development so that they satisfy the desired proposals by the people in charge of the project. In addition, for proper validation of the algorithms it is essential to follow several steps to arrive at a good way to understand it and to evaluate the current procedures.

TABLE 3.1: Minimum Requirements and Constraints

Requirements	Indicator	Constraints	Indicator
Interface between the camera and the computer	Transmission series of images to display on the computer without freezing images.	Optimum serial communication with Matlab for processing images	USB 3.0 port where camera is connected USB 2.0 port where Arduino is connected.
Store images in a video on the computer	The image is viewed in real time and stored set of images in a video format.avi	Insect Condition	To perform tests it is necessary to have the butterfly alive to obtain the images. The butterfly lasts for 10 days to perform the analysis of their compound eyes. Flies last 2 days alive.
Optical System	Camera with computer connection (with a resolution of approximately 1280 x 1024 a) to capture facets of compound eyes		
Obtain a gradient of difference between focused and out of focus measures	Gradient >0,55 with respect to normalization in base one.		
Get focus measure with each of the 5 algorithms in each of the 4 channels	Unit of Measurement of the FM is lumen (lm)		

3.1.1 Solution 1: Full custom autofocus algorithm

By solving the autofocus part that seeks the sharpest image (in focus) within a set of images captured with the GRACE goniometer, a programming environment designed to meet the needs of the device is proposed, in other words, a totally new design.

The autofocus algorithm fully adapted should include a goniometer easy to use for the person handling the machine producing a time-optimized device, a flexible program which allow an integration of a focusing algorithm that is able to receive some adjustment such as the camera parameters that is entered into the system by means of an easy graphic interface with the user, since these parameters can vary due to external factors.

This would create fully specialized software that includes details such as selecting the insect species, considering its type of pupil, including the variables of illumination, size of the image to be processed, among others, for the purpose of perform a totally original solution for the autofocus algorithm for compound-eye insects. The proposal would involve enough meeting with researchers, research and time to achieve an algorithm capable of performing a deep analysis of the compound eye.

3.1.2 Solution 2: Autofocus algorithm using frameworks for image processing

Analyzing image processing programs on the market, a library called Open Computer Vision (known as OpenCV), which works in C/C++ language, has predefined, specialized functions, methods and features for analyzing and processing images. The new development seeks an adaptation of the autofocus algorithm capable of performing the search for the best image focused within a set of images through the use of OpenCV libraries.

Libraries would help to achieve a better development of the focus as they are optimized for the purpose they have been created, this would get the set of images processed to find the best focused image. It would be an exhaustive analysis involving libraries as edge detection in order to detect that it really is the insect eye and not other forms or patterns since it is important to detect the eye section which is required to ensure an adequate analysis and avoid noise in the pictures. Finally show results of the level of focus of the image captured and that out of focus (blurry).

3.1.3 Solution 3: Autofocus algorithm adaptation

Taking into account the algorithms used so far with the GRACE device, a solution would be sought to improve them by means of modifications required by the device, this could include performing an image analysis in the four channels at the same time, to obtain the graphed focus measure as well as a standardized graph (normalized graph) for each of the proposed algorithms.

In addition, it must select which of the five algorithms had better results or if none turned out to be the most optimal, in order to always achieve the best focus image. If one of the algorithms works, then the algorithm continues to analyze which generates a focus measure (FM) faster, it should be used to define a function that is able to stop the displacement of the camera just in the moment that a high value of FM is found, that would represent the position where the image should be taken.

Apart from having normalized measures, leads us to be able to make comparisons of one algorithm with the other in each of the channels and to note the behavior of measurements graphically to note the gradient or difference between the focused and the unfocused measures.

3.2 Solution comparison and selection

Having presented the previous solutions, a comparative table has been made in order to clarify the choice of the design proposal which is adjusted to the needs of the project and has a better performance. The solution evaluations are performed as detailed below in Table 3.2, it is necessary to choose a measure that determines the proposal to be selected by means of the following evaluation:

- Very Good with 4,
- Good with 3,
- Regular with 2,
- Inefficient with 1

The evaluation of the solutions is performed as detailed below in Table 3.2

TABLE 3.2: Solutions Comparison

Evaluation Criteria	Solution 1	Solution 2	Solution 3
Functionality	4	4	4
Time Implementation	1	2	3
Cost Implementation	1	3	3
Integrability	3	2	3
Algorithm Complexity	2	3	3
Average	2.2	2.8	3.2

The Table 3.2 has shown that the *functionality* of all three solutions has a good acceptance; they are able to perform a focus search on the set of insect compound eye images. All solutions work properly for the purpose of obtaining the focus measurements of each image and say which is the best focused in its different adaptations and methods to solve.

Regarding the *time implementation* that involves the time consumed to develop the focus algorithm so that it goes into operation, Solution 3 worked best. It showed to be easily adaptable, using GRACE algorithm and software, and it takes less time than the other solutions by not starting from an implementation analysis such as solution one. Solution 1, in order to be a fully specialized algorithm for GRACE, it would take a longer period of time to achieve its implementation and favorable design. Meanwhile, Solution 2 has to take much time to elaborate when switching to a new bookstore. It would have to know the language to compile in a new software for the image processing, aside from finding a way to be adapted to GRACE.

The *cost implementation* is based in the materials, software, arrangement and all the aspects which get involve carrying out the autofocus algorithms. The Solution 1 has a very high cost taking it to get a punctuation of one, it means that a fully specialized algorithm requires a cost that implicate a design that is based on the needs of GRACE autofocus algorithm, which is why it suggests a deep search for tools that fit properly making it difficult to implement. The second solution has an average cost because it utilizes a new platform and compiler, the open CV library requires C language software where the autofocus algorithm has to fulfill the same functions that it does now in the goniometer. The third solution runs on the same software that GRACE first version has used, so its implementation is the most favorable because it is able to optimize several aspects like the color analyzes and camera control.

The *integrability* refers to the adaptation of the solution with the system that exists, in this case with the GRACE goniometer second version, the criterion is accepted by option one as it would help the autofocus process by its specialized orientation to the system. The Solution 2 is the furthest from integration due to the new compiler with the library causes more testing and more work to readjust to GRACE. The third option integrates better, able to run on the same old platform but with added procedures to improve the performance of the autofocus in the GRACE second versions.

The last criterion analyzed is the *algorithm complexity* which is the algorithmic procedure gets the focus measure (FM) and achieves the necessary programming for the objective. The solution one is more difficult mathematically because of the complexity of designing an algorithm with a new procedure that can carry a lot of study and knowledge; moreover, to find a solution that does a good measurement while having

various options in the market requires the adequate knowledge to choose which is best for GRACE autofocus.

In the case of the Solution 2, the complexity became less as a result of the image processing dedicated library. It performed very well with the focused image search. On the contrary, it demonstrated more complexity when adapting the mathematics that meet the current algorithms to this new language.

Third solution shows a favorable average due to having already performed the mathematical procedures of autofocus on the platform and is known to work well. It has to add new designs to help the algorithm to complete a satisfactory analysis where the algorithm performance consider the perspective of the focused image according to the human eye as well highlight the search of the image edge beneficial to demonstrate experimentally the veracity of the sharp image.

As discussed above, solution number 3 meets the appropriate conditions and obtains the best evaluation for the development of the design and the implementation of the functional prototype of the autofocus algorithm likewise the validation of the algorithms.

3.3 Algorithm Design

According to the adequate selected proposal for the fulfillment of the project accomplishment, here the explanation begins of the improvement autofocus system which leads toward a module description make use of Figure 3.1



FIGURE 3.1: High-level module solution of autofocusing algorithm design, own author-ship

The Figure 3.1 represent a high-level description of the autofocus algorithm where five modules describe the procedure, it starts with get an insect that means the way to place the compound eye insect, then setting camera includes the respective camera configuration. Following, the data acquisition consist in the interaction with whole control to catch data to begin with the image processing where apply the autofocus

techniques in order to obtain the focus measures and the end of the process make an exhaustive analyses in the data extraction.

3.3.1 Modules Description

The explanation of each of the modules is presented and it details the function that it fulfills within the execution system.

- **Get an Insect**

Insect selection is necessary for research analysis, insects has to be collected with compound eyes, flies and butterflies have generally been chosen which a small reserve of them should be kept in the laboratory for daily study.

A fundamental aspect is to identify the species with which it is going to work, both to characterize this type of vision as well as to notice specific aspects of the species, the species vary depending on the time of year.

The insects must be placed properly in the goniometer so as not to mistreat or kill them during the process of image acquisition analysis. At all time the insect must remain still and alive to obtain better results.

The flies are usually placed in a plastic tube with variable dimensions as to the tube diameter, reason that on one side requires a large opening to introduce the fly and on the other side only manage to come out the head and eyes of the insect. The large opening side should be covered with a cotton or plasticine to ensure that the fly does not come out, it is considered that the closure let air pass to not suffocate the insect.

Regarding the placement of the butterfly, it is carried out immobilizing its body by means of a cut straw that presses slightly its wings so that it does not move them. In both cases, it is applied molten wax, to butterflies in their heads, to flies in their thoraxes.

- **Setting Camera**

Before the scanning starts the insect eye must be aligned to the visual axis of the camera with GRACE, then the algorithm design requires initialized some camera settings,

specifically the image parameters which are indispensable for catch proper photos and can be restored taking into account the specified camera model, in GRACE second version is *F7 BayerRG12 1920x1200 Mode0*. Shutter, gain mode, and white balance are three important image parameters that are going to explain, important aspect is the laboratory lighting conditions which are a dark area.

Shutter mode sets the operation mode of the exposure (shutter), it means the image sensor exposure time. Exposure, integration and shutter words are interchangeable. Shutter needs to be high due to the laboratory area.

Gain is the amount of amplification which is supplied to a pixel, gain has to be a set parameter depending on the need because increase in gain result a brighter image but also increase noise which can affect image quality.

White balance indicates white balance adjustment which means a system of color correction to account for differing lighting conditions. Adjusting white balance enables white areas to look "whiter" by modifying the relative gain of Red, Green and Blue in an image, specifically the red and blue are the only ones can adjust.

The camera needs to be connected to the USB 3.0 because the transfer rate is faster than USB 2.0 as it is needed when is working with cameras. Also, the quality of communication between the computer and the camera must be considered to avoid a frozen image when the image processing is being carried out.

Besides, the camera position control has to be created for the GRACE second version, it needs to be moved up and down carrying the microscope. This requires knowing how the ELERO engine works, which is the signals present in the ENCODER of the motor. The motor requires voltage to move it up and down and be able to perform a control as to which distance the motor wants to displace.

- **Data Acquisition**

The data acquisition depends on a system that integrates two blocks, the camera position control and the computer data reception, the blocks have to interact in order to make a communication where data are sent without lose information.

The camera positioning requires a control that allows the camera displacement in the elevation axis. A microcontroller lets the control which is responsible for receiving signals

from the optical Encoder that sends an initial and final signal to know the displaced distance of the motor. Control must perform an automatic scrolling while the camera is capturing each image.

The second block involve the computer, specifically a program that through a algorithm in a script has to receive the data and be able to save in a structure, it has designed that structure called *Scanned* where it has to contain all the images with the format $(X, Y, channel, N)$, N is the number of the frame within the set of image, *channel* represents red, green or blue channel in the image or can be all of them and X and Y are the size of the frame. Also, the focus measure that give each of the five algorithm is stored in a structure called *Focus*, where the values of each frame are placed in different columns by each of the channels, i.e, a measure of each channel's focus.

Both blocks are necessary to integrate the circuit (hardware) that realizes the movement and the program (software) that saves the structures with the images and the focus measure (FM). The blocks communication has been made so that the data is received in a set time, in addition that it is bidirectional since the blocks must communicate even to go capturing images with the camera as well as the program goes sending the amount of counts that the motor must move.

The blocks communication has been made so that the data is received in the time that is expected, in addition that it is bidirectional since blocks must communicate so much to go capturing images with the camera so that the program goes sending the amount of accounts that the motor should be moved. The above together with the analysis of the focus measures verify if the time has come for the camera to stop and to be in the proper place to analyze the compound eye.

• Image Processing

The image processing is an important aspect described by means of the five autofocus algorithms that have been used so far. The processing requires an analysis of each algorithm with respect to its nature and obtain the focus measure(FM) that produces the algorithms, the measure has been used to have a sharpness measure which is determined by different mathematics and is calculated for the three channels in each of the algorithms, focus measure has a lumen as SI derived unit.

The higher the focus measure (FM) the sharper the image because of that it is thought to use such measurement in order to normalize (as a standardization) data obtained by each algorithm and channel to perform comparisons. Such process as explained above can be seen in the validation Section 3.4 of the algorithm.

The implementation consist of making a pre-process of the image, in choosing the size wanted for analysis, then proceed to capture a set of images which are processed by the 5 algorithms in parallel and to obtain the focus measures not only the image in its entirety but also an FM measurement in each of the channels. Previously it was not done and is required to know the behavior of each channel separately.

In addition, a standardized graphics of the parameters have to be displayed, the best complete image as well as the image in each of its channels will be displayed. Finally, through the use of edge detection techniques, the images should be evaluated in order to see the facet number of the compound eyes.

- **Data Extraction**

Once the process has finished acquiring the necessary data set by the algorithm designed, the information must be manipulated to give interpretation and give an explanation. For this purpose, the data has been classified into several types of graphs, tables and tables representing the data.

The graphs show the data of the focus measures for each algorithm and channel, both in its original form and in its standardized form so as to be able to analyze which is the channel with a better performance, which was the fastest and which algorithm had higher FM values.

The tables have been designed to show images according to the algorithms of edges detection in order to highlight the best results. Finally, the tables show images with the algorithms that were favorable and not so favorable.

3.4 Validation of Algorithms

In order to give mathematical validation to the algorithms it is fundamental to give a broad explanation for their understanding, so in this procedure it is carried out by the

elaboration of several stages to follow which are detailed below.

- Execute algorithms: is done by running the five algorithms in Matlab software, note the procedure that follows, find out what each function does, and represent the data in graphs by channels of each algorithm to notice its behavior, more detail see Section 5.3.
- Nature of algorithms: perform a search for papers that explain the nature of the five algorithms approach, explain their mathematical foundation as to why they are able to get a measure of sharpness with their procedure, to get more information on a broader scale go to Section 4.3.1.
- Normalize Algorithms: the algorithms of autofocus need to be normalized to obtain a base of comparison, in each algorithm is taken the greatest measure of sharpness reached and with this value is normalized the other images. The procedure is going to be done for each of the channels (Red, green, blue and grayscale), it is presented in Section 5.3.1.
- Comparison: to have the algorithms normalized by obtaining graphs with the best of the best images of each of the comparisons as well as some graphs with the measure of focus against the number of frames, comparison is explain in more detail in result Section 5.3.

Chapter 4

Methods and Materials

Methods and materials section has been dedicated to explain the software level of each one of the modules describing how the autofocus procedure has been done in Section [3.3.1](#).

The algorithms operation have been described in a pseudocode mode in the corresponding appendix which is indicated in the appropriate context. The algorithms have been used fully and they are explained in a high level procedure for an understanding of the user.

Also, the materials have been detailed to match with the project methods and the insects that have been employed are at the end of the chapter.

4.1 Setting camera

The camera configurations have been achieved by previous research, but for this occasion in the implementation of GRACE version two, to have an appropriate order to obtain images just as required, it has been planned to have an order as described the procedure shown in the Figure [4.1](#).

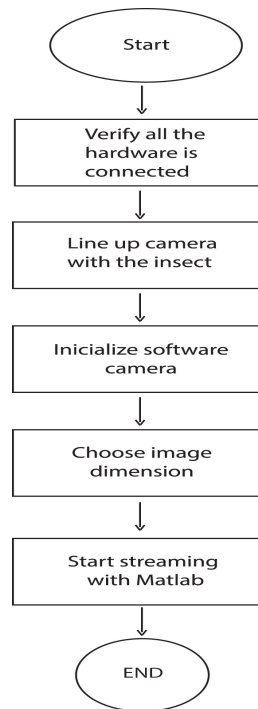


FIGURE 4.1: Camera configuration process diagram, own authorship.

According to the diagram in Figure 4.1, configuration begins with verification that all hardware is properly connected, it includes the cables connecting to the power outlets and the connection of the camera to the USB 3.0 port. Then, it must be necessary to align the camera with the eye of the insect in order to obtain a good reference of image with which to proceed to make the image that it is showed in the following Section 5.1.

The third step is to initialize the camera software, that refers to the adjustment of the required image parameters of the camera, the pseudo code design of setting camera has been described in Appendix A. The image parameters to be placed in the initialization software script are preliminary product values that have been the same values in the last investigations. These are Shutter with a value of 16.65, a gain of 24 and a value of red and blue of 589 and 901 respectively for white balance.

Then, the dimensions of the image are chosen, this will be done to have a limited region of 200 x 200 for an appropriate selection which is needed to process the image, this process is explained in the Appendix A. The size aspect had not been chosen before, but this was essential to add as it was noted that it helped to only include in the image the insect composed eyes and thus avoid having any part of the insect skin and cause wrong results. The size selecting algorithm is part of the auto-centering algorithm that has been done by an industrial engineering student [7].

Finally, the streaming video has to appear in the screen, to show the region of the image that it use to analyze. The video has had certain problems since the camera is moving the image is not continuous and rather remains in a frozen state while updating the image after the movement. After several studies, it was determined that the difficulty has been solved by means of a connection system called attention of interruptions, attending and transmitting the request to display video continuously without affecting the other functions of the machine.

4.2 Data Acquisition

As mentioned, data acquisition involves the blocks interaction between the camera position control and the software where the image processing is performed. Figure 4.2 shows the procedure which represent the design of blocks execution order.

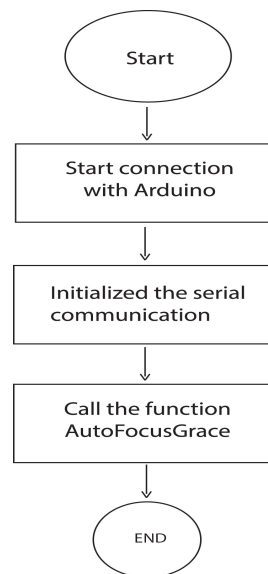


FIGURE 4.2: Data Collection process diagram, own authorship.

The acquisition begins with the connection of the microcontroller with the others electronic parts, entails the respective hardware configurations between the DC motor, the Encoder, the motion system and the Arduino connection, not the computer, as shown in Figure 4.3.

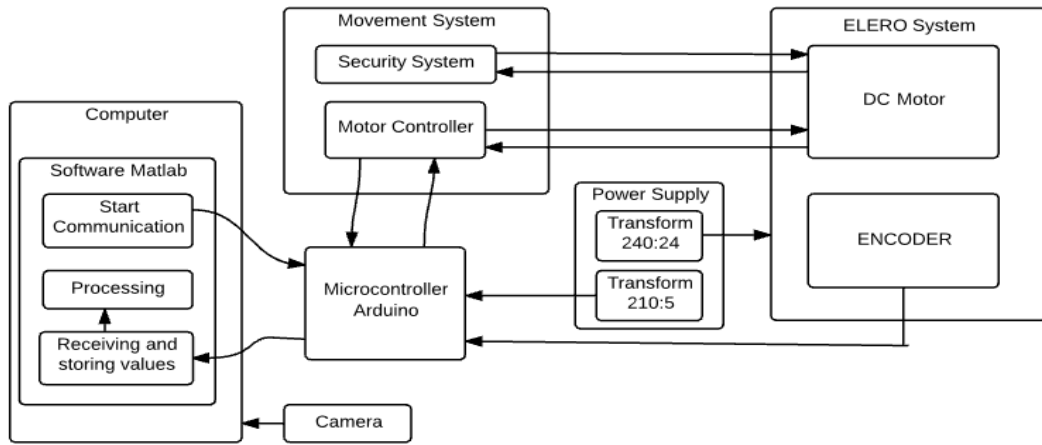


FIGURE 4.3: Data acquisition system showing the two blocks that comprise them, computer data reception shown in the left block and the camera positioning circuit represented by the other blocks to the right.

The Arduino connection design has been depicted in Figure 4.4 to give the sequence how to start the connection and the data flow. It is important to mention that this happens when the Arduino is in the automatic condition, if it is in manual condition, it means, the camera position moves according to Up and Down switch that are in the external hardware.

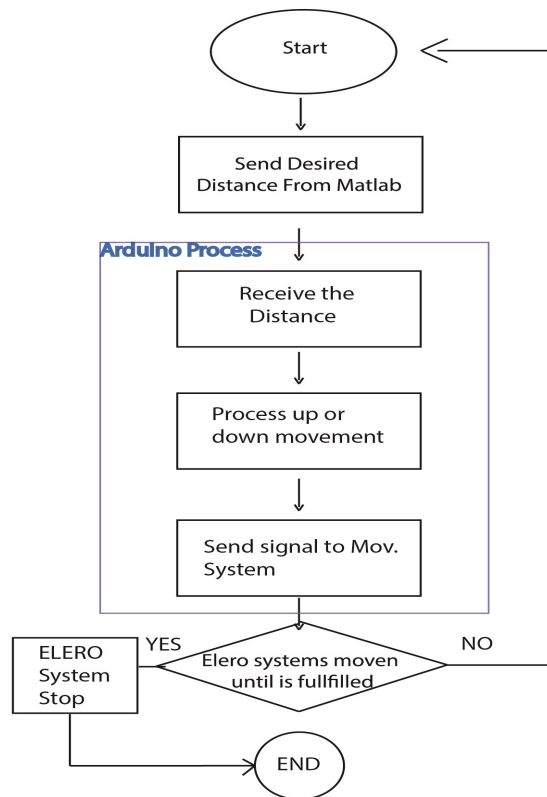


FIGURE 4.4: The Data flow of start communication with Arduino in automatic condition, own authorship.

The diagram of Figure 4.4 starts with specifying the desired distance from Matlab, the distance would be specified by calling a function to move the motor in required place or user can specify the photos number that is needed and the algorithm proceeds to move one appropriate distance to take photos, the algorithm that includes the above describe movement function is explained further at the *AutoFocusGrace* function.

The distance has been established in string format to the Arduino which receives the distance to encode it to an *integer* format, where it is known if the movement is up or down, it means that the sign distance is positive upwards and if it is negative goes down. Then, the Arduino system sends three signals to the movement system, which are the EN, IN1 and IN2, they are indispensable to enable the movement.

EN is set that it is always active to move, IN1 is *high* and IN2 is *low* meaning going upwards, whereas IN1 is *low and* and IN2 is *high* signaling downwards. The movement system block was designed by Dr.-Ing. Mauricio who adapted a motor protection system from overcurrents.

Once the motor controller receives the signals, it causes the ELERO DC motor to move in the set direction until the requested distance is reached, at the beginning and before stopping, the motor sends the position through the encoder to verify that it is the desired distance and then stops and waits for the next instruction.

Initialize the serial communication is the follow step according Figure 4.2. The initialize is carried out by the Matlab software and has been established to begin the communication between the circuit and the autofocus algorithm, it is an important procedure that must guarantee a good data reception without losses, for it the initialization sets up certain parameters that synchronize the communication which are the data bit size, stop bit, baud rate and timeout, these values are in the Appendix A (List 4.3) where is the code.

When the system is completely ready with all the aforementioned connections and configuration, the last step consists of the called *AutofocusGrace* function in which the algorithm has received the video variables, the serial communication is available with the circuit to move along the axis and acquiring an image set that is captured on certain number of pre-established steps are starting to set.

In addition, the five autofocus algorithms execution is happening while good position is calculated with the values that result of the algorithms which indicate where a clear image can be taken and, when finding the adequate displacement the goniometer stops.

The list code in Appendix A shows the *AutofocusGrace* function pseudo-code, is important to mention that the images have been stored in a structure as well as each value of focus measure generated by each of the techniques, this to analyze the data after the process.

4.3 Image Processing

The image processing includes the part of the mathematical explanation of each of the algorithms as well as the search of which is the most appropriate technique to achieve clear images and that the camera position stops when it finds a maximum value of focus measure and display the live image.

The algorithms explanation has been an important contribution to the development of the goniometric device because the algorithms were only implemented experimentally but did not have a basis for which it was given to the task of searching several studies to compile the information that summarizes the most relevant of each algorithm.

4.3.1 Algorithms Explanation

The presented algorithms are able to automatically measure the sharpness of focus in an image. The sharpness measure underpins and validates the Energy of Gradient, the Energy of Laplacian, the Gaussian Derivation, the Tenengrad, and the Vollaths Correlation algorithms. They must give a metric of sharpness. All the methods implement the idea of searching focus measure quality which help to represent to high frequency content in the image, in the most of the cases.

The autofocus function is intended to give a value indicating the degree of focus of each image considering the different information contained in focused and unfocused images. It is considered a blurred image the convolution between the image and a Point Spread Function (PSF) that generates a decrease of the high frequencies in the image.

4.3.1.1 Energy of Gradient

The gradient energy is a method that achieves a measure of focus by analyzing all the pixels of the X and Y axis of a frame or image to be analyzed. The gradient provides a vector which points in the direction of the greatest change relation of the function f at the location (x, y) [16].

The vector magnitude from a continuous function of the image $f(x, y)$ (2.1), provides the change ratio per unit of distance in the vector direction. In order to detect changes in the levels of Gray gradients occurring in degraded areas or regions, the gradient [19] is expressed as follows in:

$$\sum_{x=1}^X \sum_{y=1}^Y (g_x^2 + g_y^2) \quad (4.1)$$

where g_x and g_y are the first-order difference of the image calculated along X and Y direction (X and Y represent the image dimensions) as they are explained in the following expressions:

$$g_x(x, y) = g(x + 1, y) - g(x, y) \quad (4.2)$$

$$g_y(x, y) = g(x, y + 1) - g(x, y) \quad (4.3)$$

It can be noticed that an difference in the intensity of adjacent pixels is calculated in both the X and Y direction, resulting in a scalar value which is sensitive to noise. To combat noise, an autofocus algorithm has been implemented in smoothing the image by calculating the gradient mean (4.1) obtained.

In general, the principle of the Energy of Gradient is a differentiation-based focus function. It is to consider the existence of an edge if there is a discontinuity in the image intensity function. It means that a maximum is obtained in the derivative that is applied when searching for the intensity values of the image.

4.3.1.2 Energy of Laplacian Autofocus Algorithm

The Laplacian is a method to measure the amount of edge present in image, using the second derivative for image sharpening. Technique consists of establishing a discrete formulation of the second order derivative and then constructing a filter mask based on that formulation.

The isotropic filters refer to response from the discontinuities in the image, independent of the direction the discontinuities come from. It is called an isotropic filter a value that does not changed if a rotation occurs in the image. The filter gives the same result regardless if applying filter to the image first or after rotating, it means is a rotation invariant [16].

The Laplacian is the simplest isotropic derivative operator for an image function $f(x,y)$ of two variables (x,y) which can be shown as:

$$\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} \quad (4.4)$$

In (4.4) serves as the union of the image with the filter mask that performs the method to achieve the focus measure value. Therefore Laplacian takes into account the second derivative of the image function $f(x,y)$ given that it is a property suited for sharpening image and it is easy to implement. The derivation becomes partial with respect to x and y to consider both the horizontal and vertical pixels. The Laplacian derivative is a linear operator that need to adapt in a discrete form, it uses the following definition in the x direction and in the y direction:

$$\frac{\partial^2 f(x,y)}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y) \quad (4.5)$$

$$\frac{\partial^2 f(x,y)}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y) \quad (4.6)$$

The preceding (4.4),(4.5) and (4.6) are attached for the Laplacian discretization as show in:

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y) \quad (4.7)$$

The implementation (4.7) uses a filter mask (4.8) which gives an isotropic result for rotations in increments of 90 degrees. The filter mask is proportional in a average of the intensity level of a pixel contained in the surroundings of the filter mask. The process is able to smooth the image for the purpose of reducing noise [16].

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (4.8)$$

Laplacian is a derivative operator which is used to highlight intensity discontinuities in an image and neglects regions with low intensity levels, leads to generate clear images. The method uses the square of the results of the convolution of the image with the mask of the filter mentioned above, a data media is obtained and also the focus measure to analyze the sharpness of the image.

4.3.1.3 Gaussian Derivation Autofocus Algorithm

It is a focus method that results from combining a smoothing and a gradient to measure the energy contained in the image [19]. The level of smoothing is used for the medium of Gaussian Derivation application, while the gradient is provided by the mask that is chosen according to the desired purpose.

The Gaussian Derivation operator is implemented by the use of the convolution between the image function and the first derivative of Gaussian, as represented below [20]:

$$F(\sigma) = \frac{1}{NM} \sum_{x,y}^{X,Y} [f(x,y) * G_x(x,y,\sigma)]^2 + [f(x,y) * G_y(x,y,\sigma)]^2 \quad (4.9)$$

where $f(x,y)$ is the image gray value, $G_x(x,y,\sigma)$ and $G_y(x,y,\sigma)$ are the first-order Gaussian derivatives (probability distribution) in the x and y direction respectively, NM is the among of image pixels and sigma represents the scale of prominent features. It formulae can be reduce as:

$$F(\sigma) = \frac{1}{NM} \sum_{x,y} f_{x^2} + f_{y^2} \quad (4.10)$$

where f_x and f_y are the derivatives at sigma scale σ in the x and y direction. The sigma of the Gaussian filter should be chosen in a way that noise is maximally suppressed. Because of the Gaussian filter used is

$$G(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (4.11)$$

The filter (4.11) eliminates Gaussian noise as well as detects edges, which helps to define this technique. Gaussian Derivation method explained on this criterion, gives a response to the best focus position when finding a local maximum, in this way it can be said that it is a complex method that works for the search of the sharpness of the image when implementing Gaussian filters in which the results with scalar value indicates the focus measure that has been achieved.

4.3.1.4 Tenengrad Autofocus Algorithm

It is a technique that analyzes the nature of the edge based on the differentiation of the image, that is to say, when there is a distortion (blurring) between dark and clear boundaries in the images. Because PSF (Point Spread Function which is a response to the impulse) is distributed at each pixel intensity, we take into account the differentiation of the image to find out the measure of focus. Algorithm performed by Tenenbaum [23] is an edge detector for autofocus, which is described in the following steps:

1. Set the contents of a register R to zero.
2. Evaluate the Sobel operator S in (4.12) at each pixel p in the processing window to get $S(p)$ (4.13).

$$i_x|_p = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} i_y|_p = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (4.12)$$

$$S(p) = \sqrt{i_x^2 + i_y^2} \quad (4.13)$$

3. Compare $S(p)$ to a threshold T .
4. If $S(p) > T$ add $S(p)$ to R .

First step, the register R is where the focus measure scalar is saved (in our algorithm we call that record FM) after the pixels have been processed. Second step, the Sobel operator is used to detect edges and gives an estimate of the magnitude of the gradient intensity of an image, leading to a processing of each pixel of the image [21].

The Sobel operator results in a gradient vector corresponding to the norm of the vector which is the result of crossing by an edge whose direction is from the darkest to the lightest points, instead of passing through a region with constant image intensity in which case the vector is zero [24].

Finally, as the third and four steps say, the edges that are taken into account for the sum of the gradients are larger, have an appreciable height, a threshold (T) to use which transforms from the algorithm to nonlinear (nonlinear is due to the spatial mask that involves squaring and square roots and whose discussion can be found below) but functional for Tenengrad algorithm.

- **Threshold (T) Selection**

The threshold value could be chosen arbitrarily to be zero, it does not cause any problem but it is not convenient and it is inappropriate to do so in this way, which is described below as its value is determined. The threshold parameter (T) has been treated as an adaptive parameter [23] as it changed from a high value when it is possible to a low value when is necessary [25], based on the above, the threshold has been chosen to be 75% of the maximum value of the focal length. Making the thresholds larger when the image comes into focus causes the threshold to continue and eliminate the texture of the background.

The threshold should be set at the beginning of a focus run, so that the value can be based on an evaluation of the noise in the gradient domain. This means that there is a reduction of the noise when using the temporal average, when the variance of the noise

is known and also when considering that the untreated image has a normal noise with mean of zero and variance (σ_n^2) [24].

- **Mathematical Description**

Mathematically, the desired gradient is the intensity function of the image represented in a two-dimensional vector given by two variables, the components of which are given by the partial derivative estimates i_x and i_y in the horizontal (x) and vertical (Y) respectively [16]. To obtain derivative approximations, two 3x3 Sobel Kernel operators are used (shown as i_x for horizontal axis and i_y and for vertical axis in (4.12) which convolve to the original image [26].

With the filter is possible to take the horizontal and vertical components to squared them and sum all the products of the vector of each pixel of the image in order to generate a spatial mask with the pixels neighborhood, which is a linear combination [16]. It is required to combine the horizontal and vertical gradients like in (4.13) to calculate the magnitude of the gradient, (i.e, to achieve the measure of focus that gives Tenengrad algorithm) [23].

It is important to mention that convolution plays an essential role in image processing, since it involves imaging and other image processing operations, convolution helps where lines and dots are blurred, resolution is reduced [26]. The convolution is a mathematical operator that transforms two functions, for instance A and B, that generates C which is a representation of the magnitude in which A is superimposed and a translated and inverted form of B is generally said to be a statistical type of moving average, i.e, it is an analysis of a set of data in the form of points that give rise to averages of the data to be analyzed. In other words, it means that the gradient vector leads to the maximum possible increase of the intensity, and magnitude vector represents the intensity change as aforementioned.

4.3.1.5 Vollaths Correlation Autofocus Algorithm

The Vollaths Correlation method looks for the similarity between a central pixel and a predefined zone (surroundings), it is an image relation with itself, this is spatial filtering processing when using the autocorrelation operation. Autocorrelation is based on

cross-correlation, which means that it looks for the correspondence of a selected area in the image with respect to another image (or with the same image) to determine its displacement [26]. Vollath proposes a method that is based on experimental determinations using several criteria and properties until finding the criterion that satisfies the search for the value of the focus measure [22].

The correlation criterion found by Vollaths is shown below

$$\underbrace{\sum_{i=1}^{M-1} \sum_{j=1}^N g(i, j) \cdot g(i+1, j)}_1 - \underbrace{\sum_{i=1}^{M-2} \sum_{j=1}^N g(i, j) \cdot g(i+2, j)}_2 \quad (4.14)$$

where $g(i, j)$ mean intensity at the point (i, j) and first and second part are explained as:

1. Take the sum of all intensity pixels in the coordinates (i, j) multiply them by the next neighbor next in x while retaining the position in j ,
2. The above description (step 1) subtracts the sum containing all intensity pixels in the coordinates (i, j) multiplier by the neighbor in x two places after and conserving the position in j .

Properties

In order to be able to analyze the formula one has to take into account two important properties, one is with respect to the contrast and the other with respect to the influence of the focus scene. In addition, certain autofocus criteria are defined and defined below [27].

- **Constrast**

The propertie in respect of the constrast is definite as the differences in brightness of two neighbouring points, the constrast is weighted by the amplitudes [22]. The local intensity $g(i, j)$ is decomposed into:

$$g_{i,j} = a_{i,j} + \mu + r_{i,j} \quad (4.15)$$

where $a_{i,j}$ is the distribution of amplitudes with the property, μ is the mean value of the brightness and $r_{i,j}$ is the noise component. Also, (4.15) have been derived with the properties that is defined then:

$$\sum_i \sum_j a_{i,j} = 0 \quad (4.16)$$

$$\sum_i \sum_j r_{i,j} = 0 \quad (4.17)$$

$$\sum_i \sum_j a_{i,j} \cdot r_{i,j} = 0 \quad (4.18)$$

$$\sum_i \sum_j r_{i,j} \cdot r_{i+1,j} = 0 \quad (4.19)$$

The (4.16) - (4.19) are assumed to achieve the definition of Vollath, They are summarized in the sumatory of all distribution of amplitudes are zero in (4.16), the sumatory of all noisy components is likewise as the property holds in (4.17). The amplitude function and noise are not correlated in (4.18). The noise in neighboring pixels is not autocorrelation. Additionally, the following general assumptions are important:

$$\sum_i \sum_j a_{i,j}^2 \approx \sum_i \sum_j a_{i+k,j}^2, k \in \{0, 1, 2\} \quad (4.20)$$

$$\sum_i \sum_j r_{i,j}^2 \approx \sum_i \sum_j r_{i+k,j}^2 \quad (4.21)$$

$$\sum_i \sum_j a_{i,j} \approx \sum_i \sum_j a_{i+k,j} \quad (4.22)$$

Using the above properties and the general assumptions, one can recognize the influence exerted by the contrast in (4.14). The growing sharpness of the image makes both the amplitudes and the contrast becomes greater leading to an increase in focus. It is important to mention that ration of the noise free that intervene in (4.21) does not affect the procedure.

- **The scene on focusing**

The character of the scene is important in view of the distribution of amplitudes $a_{i,j}$ since now the intensity $g(i, j)$ is decomposed as

$$g_{i+k,j} = a_{i+k,j} + \mu + r_{i+k,j}, k \in \{0, 1, 2\} \quad (4.23)$$

where the next neighbor amplitude is now taken into account, keeping the axis j constant. In addition, with (4.23) and general assumptions and definitions are replaced in (4.14) and get:

$$F_4 = \sum_i \sum_j (a_{i,j} \cdot a_{i+1,j} - a_{i,j} \cdot a_{i+2,j}) \quad (4.24)$$

Note that the criterion (4.24) is still unaffected by noise. This is because the small differences in brightness generated in the brighter or darker parts of the image are taken into account in the same way as the large differences that arise in case of small amplitudes [22].

Besides, average brightness is considered in fine structures (in small differences in brightness). As observed in (4.24) it does not refer to the brightness so we interpret the parameter at the maximum amplitudes $a_{i,j}$, (i.e, the largest values of amplitude obtained). To analyze the dependence of the frequency, the distribution of amplitude is based on Fourier series where $a_{i,j}$ is:

$$a_{i,j} = \sum_{i=n} b_{i,j} \cdot \sin\left(\frac{n \cdot \pi \cdot i}{N}\right) \quad (4.25)$$

where n belongs to $(1, 2, 3, \dots, N)$ and $b(i, j)$ represents binary image in the conversion between analog to digital, is an assigned gray value with the intensity $g(i, j)$ where the probability of a logical 1 at point (i, j) on an average is the brightness. With analysis applied to the (4.24) is obtained:

$$F_4 = \sum_j \sum_n b_{n,j}^2 \cdot \left(\cos\left(\frac{n \cdot \pi}{N}\right) - \cos\left(\frac{2 \cdot n \cdot \pi}{N}\right)\right) \quad (4.26)$$

It is observed that frequency-dependent average is more complex, but it can be recognized that (4.24) considers the high frequencies to give a brightness distribution which generates a maximum sharpness value. According to the influence of the scene on focus (4.14) the maximum image sharpness is reached for a scene influenced by strong contrast [27].

- **Autofocus criteria**

For the application of the Vollaths correlation algorithm it is important to consider four criteria:

- Expenditure in terms of computation: the implementation of (4.14) do not requires extensive computations.

- Working range: it means the defocussing zone within which correct focusing is still possible, to obtain a high working range, low frequencies are taken into account, but since the (4.14) works with high frequencies, the reliability of achieving maximum image sharpness is not reduced.
- Image sharpness: according to the (4.14) an image is deemed to be sharp when obtaining the maximum amplitudes.
- Sensitivity to noise: the process of (4.14) avoids the noise by advantageously selected algorithms, because the maximum amplitudes are considered for the process of approach of Vollath, since this technique by its procedure of correlation covers a greater range of amplitude in which the calculation of the focus measure.

The Vollath's correlation is a method which reaches an optimal manner to provide the focusing in images and to avoid the noise disturbance when taking into account the maximum amplitudes in the process when analyzing the image, the procedure shows by scalar value the focus measure when obtaining an average of the analysis of the untreated image multiplied by the differences of pixel intensities.

4.4 Materials

It is presented a Table 4.1 with the main materials that integrate the hardware system of the camera movement for an adequate autofocus in the macroalgorithm.

TABLE 4.1: Main hardware materials list for the GRACE goniometer

Item	Symbol	Description	Quantity	Brand	Model Number	Price (USD)
Arduino	Arduino	Microcontroller used to control the communication between the hardware	1	Arduino	Arduino Uno	26
ELERO	Elero	System that contains a motor of DC and an encoder to be able to know the displacements of the camera that is supported by this system.	1	ELERO Junior	Junior 1 version 3	450
Camera	camera	Camera for image capture compatible with Matlab and good resolution (1920x1200 pixel with 41 FPS)	1	Point Grey Fly	BFLY-U3-23S6C-C	495
Motor Controller	Mov. System	Bridge H which helps control the rotation direction of the DC motor	1	Okaphone	L298N	8

The Table 4.1 describes the main materials used for the hardware connection presented in Figure 4.3. The materials have been selected for their performance that favors a good application for the GRACE second version, since the interaction of programs that contain the algorithms with the selected equipment allows the data flow to be properly.

For example, the above is evidenced in that the value of the distance to move the motor that transmits the *AutoSlope* function is sent correctly to the equipment generating the displacement. The displacement is given once the function signal reaches the Arduino control that performs its work thanks to the internal programming that has been configured which sends signals to the other equipment as the Elero system and the Motor controller so that it is the desired the camera movement.

One material that is indispensable for the project are compound eye insects, in this occasion flies have been used. Flies are captured around the laboratory or provided by the biology school at the University of Groningen. The species that base the tests performed have been chosen due to the availability of the season which are *Sarcophaga sp.* depicted in Figure¹ 4.5, *Scathophaga stercoraria sp.* showed in Figure² 4.6 and *Calliphora sp.*



FIGURE 4.5: Front view photo of the *Sarcophaga* fly which is appreciated the head and the compound eyes, photograph taken by H.L.Leertouwer.

¹ Photo taken by the collaboration of Hein.L. Leertouwer who is a researcher in Computational Physics Laboratory at University of Groningen.

² Acquired by H.L.Leertouwer

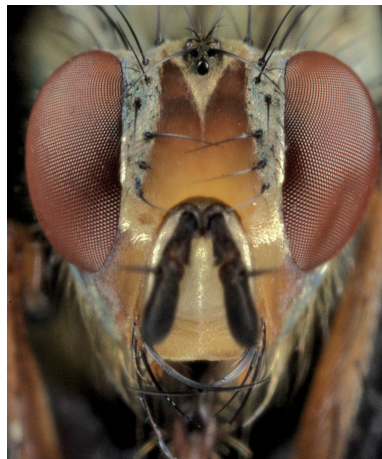


FIGURE 4.6: Front view photo of the *Scathophaga* fly which is appreciated the head and the compound eyes, photograph taken by H.L.Leertouwer.

Chapter 5

Results

Chapter shows the tests to evaluate the design previously discussed. An aspect to emphasize is that the image capturing has been carried out with several species of flies. However, the last tests have been realized with *Sarcophaga sp*, *Scathophaga stercoraria sp* and the third fly is believed to be most likely *Calliphora* since the fly has not being as closely examined as the other two. The genus of the latter has been determined by the darkness of its body.

5.1 Setting camera results

As the camera configuration procedure was designed in Section 4.1, the following results have been generated when the camera was aligned with the insect. Figure 5.1 is able to show a full size of the compound eye in the computer screen.

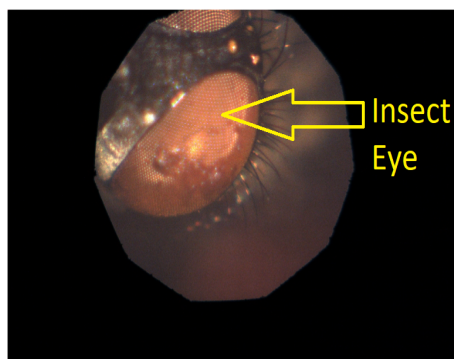


FIGURE 5.1: Format showing the live video that is observed in the computer screen, representing the full size compound eye image without the desired region selected, took to [7].

At this point, it is when the camera has already been initialized as it was designed and the center point of the image in the region desired by the user has been selected, the result of this selection is shown in Figure 5.2.

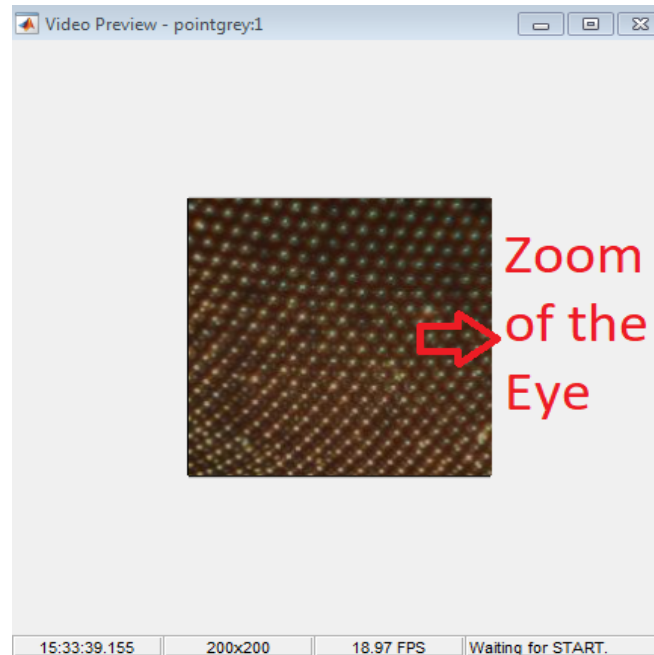


FIGURE 5.2: The image zoom format of the insect eye is illustrated on a specific area. The size corresponds to 200x200 pixels, own authorship acquired from Matlab software.

The desired region size of the window has been established by means of an algorithm implemented by laboratory partner [7] who solved the problem choosing a point of interest within the image where it is visualized what is wanted, a box of 200 pixels is established upwards, downwards and sides around the selected point.

The video transmission had several problems for not having a continuous streaming, so it has been solved by incorporating in the camera initialization script a section where it includes the interruptions as found in Section 4.1. The video streaming which is obtained is 19 frames per second (FPS) on average. The implementation of the interrupt has generated a command line as shown:

BytesAvailable event occurred at 13:01:29 for the object: Serial-COM12.

The command is generated in the command line of Matlab, which indicates that an appropriate and continuous connection is given since the transmission of the video is taken care of when requested.

5.2 Data acquisition results

Having achieved continuous video streaming, the system communication between the Matlab scripts containing the algorithms and the camera positioning circuit must transmit and receive signals without loss to know where the engine is moving as Section 4.2 has established.

An important aspect is both the Arduino code and Matlab script have to be set with the same number of baud rate which in the project the number of signal units per second is 9600. The other parameters are set, these one are the data bit representing the data size that is 8 and the time out that is the waiting time to transmit or send data is 60 seconds.

After the configuration serial parameters, the communication begins sending the desired distance which is interpreted in the Arduino microcontroller. The camera displacement has been made automatically to go capturing a certain amount of images and go processing them with the techniques of autofocus in order to have a synchronized process.

The *AutoFocusGrace* function has been created to fulfill the synchronization process that despite having a good performance, then has been implemented a servo control based on image that is known as *AutofocusSlopes* function which apart from doing the same functions as the previous function also complies to stop the camera displacement until an image satisfies the focus conditions have been set in the function.

The *AutoFocusGrace* function consists of making a camera shift to capture several photographs, analyzing with the five autofocus techniques to finish with a return to the position where it had the best approach according to one of the five algorithms as you can see in the Figure 5.3.

According to Figure 5.3, focus structure receives the *the video and leo* input parameters that represent the signals which make sure the live video of the insect eye. The *leo* parameter is who communicates Matlab with the Arduino, the function has started after *leo* has been received.

The cycles number is the number of images that is acquired in the execution of the algorithm, the delta number is the number of counts with which the camera position has advanced and each one of the images taken.

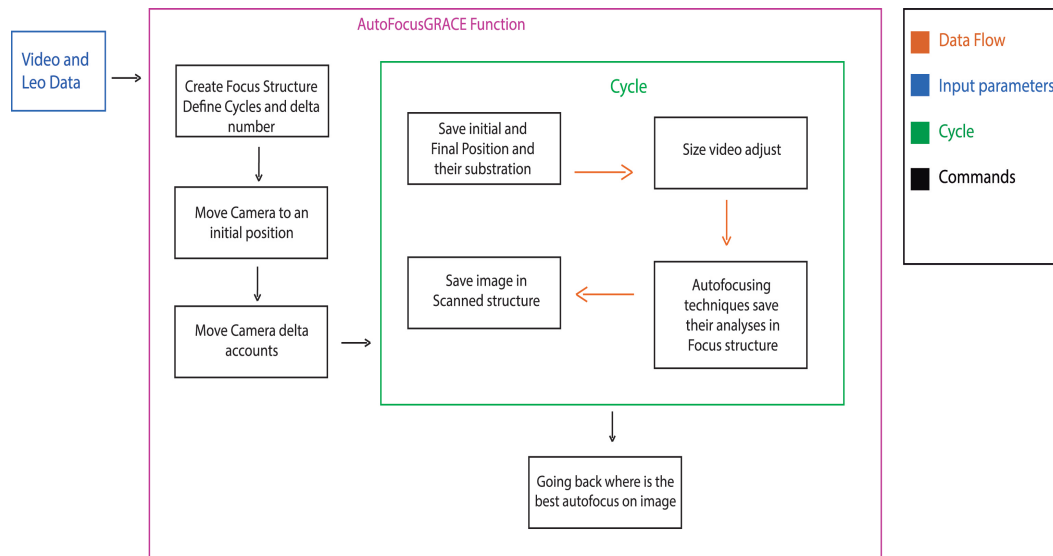


FIGURE 5.3: Block diagram of the AutoFocusGRACE Function part of the Data Collection automation program. Four different colours, the red arrows represent the path taken by the data flow, the blue box represents input parameters with the function needed to run the script, the green box represents the cycle that repeats until the number of Cycles are defined by the user and the black boxes are the commands to execute the algorithm, own authorship.

It begins the cycle saving in the Focus structure the initial and final position of each image that has been taken. The image taken has been adjusted to a certain size to be evaluated by the five autofocus algorithm, the techniques are able to be evaluated in the three channels and their FM values is stored in Focus structure (more detailed in section 5.3).

Also, the captured image is stored in the Scanned structure which has been used in case the researcher wants to analyze or perform further tests outside the function. The cycle ends base on the number of cycles a set values at the beginning of the algorithm.

It is important to mention that the Focus structure is a matrix with columns and rows, where each column represents a measurement type and the row stores the measurement corresponding to the evaluation of each image, while Scanned structure is an array where stored the size and the three channels of the each image.

Finally, *Going Back()* is a function which intend for the camera to return to the position where the best approach was found during the sweep of the movement, but it did not happened. The camera has returned to the position it wanted.

The motion counts of each image captured had an error of 3 more values, which means that if one put a measure W (where W symbolize any value), the camera moved that measure W plus three accounts (later that situation is reported at the end of this section).

Not having been able to return to the position led to decide for a function that would allow to acquire the images, analyze them, store the results and stop at the point of finding the best focus

Here is where the *AutoSlopes* function comes in. Function consists of performing the analysis of the image and likewise stores in Focus structure the FM values that arise from each autofocus technique, but now it takes groups of five FM measurements and calculates the slope through evaluating the stop condition which consists whether the value is positive and less than the previously calculated slope value. The last step is done in order to know if the highest value of focus has been reached and then to stop the displacement of the camera and the image capturing.

The AutoSlopes function part corresponding to the slopes calculation has begun to work after five values of focus measure (FM) have been taken to make the first slope. And so that the stop condition is carried out, it has been arranged that ten focus measures have been passed so that two slopes have been taken to assess the condition. In figure 5.4 is showed the block diagram of the AutoSlope Function.

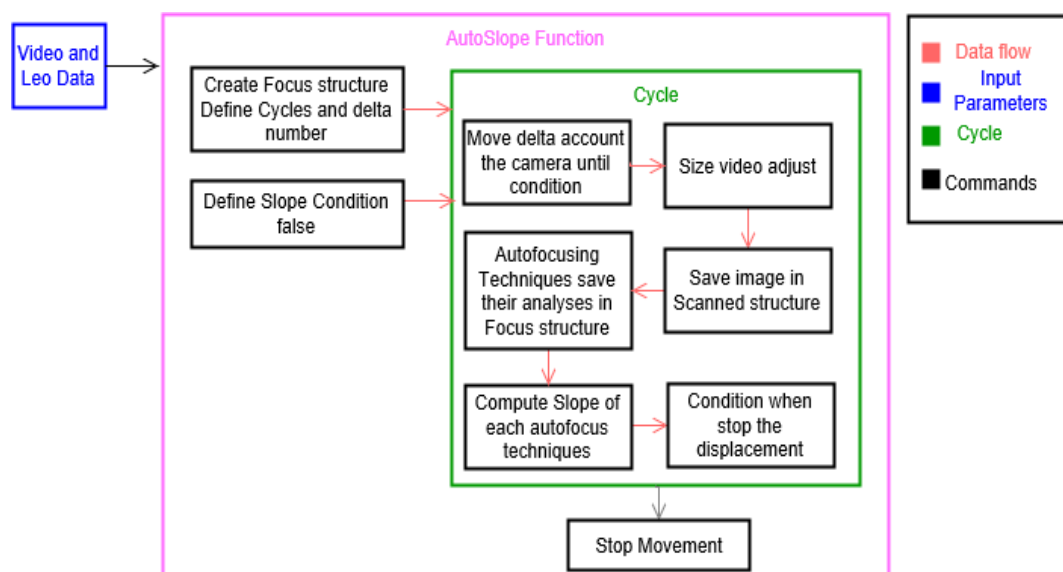


FIGURE 5.4: Block diagram of the AutoSlope Function part of the Data Collection automation program. Four different colours, the red arrows represent the path taken by the data flow, the blue box represents input parameters with function needs to run the script, the green box represent the cycle that repeats until the condition is reached and the black boxes are the commands to execute, own authorship.

AutoSlope function started by receiving the *leo* and *vid* parameters, which are the same as last function, the number of cycles is continued, delta is value defined in the program and a *SlopeCondition* is a boolean variable which is set to be equal to false.

Then, the cycle began where the camera has been adjusted to a suitable position before the movement started each delta counter. The image had been adjusted to the appropriate size and the image has been saved in the Scanned structure. The camera continued evaluating the image in the autofocus techniques that kept FM value in the Focus structure.

The process continued the calculation of the slopes taking five FM values with a proper function of Matlab called *Polyfit*; Polyfit function has been applied with each of the autofocus techniques, this function has been useful for calculating the slopes.

As mentioned, once a ten focus measure and the first two slopes have been calculated, the stop condition must be evaluated which if it is reached the *SlopeCondition* variable is changed to true boolean value because the FM has been fulfilled, here the cycle ended and the movement has stopped.

For both *AutofocusGrace* and *AutoSlopes* it has been necessary to start from a position where the eye of the insect could be seen, before the displacements were initiated, besides making sure that the insect is alive for the analysis, that two functions are in Appendix A(List 4.4 and 4.5).

The delta count number has been sent from Matlab as a string variable to Arduino where the parameter has become a integer variable to be understood by the Arduino code. If delta parameter has been positive the camera went up but if it has been negative the camera went down. The rotation direction has done by means of the motor controller integrated circuit where the sign finally arrived to Encoder system.

Encoder has moved until the condition in the Arduino code which consist in the subtraction of the final position minus the initial position has been subtracted from the delta signal sent and, still, this result had to be less than a magnitude of two. The magnitude has been chosen by the necessity of accuracy.

The previous Arduino condition has been achieved after several tests when the movement of the camera could be stopped just when it has wanted. It is assumed that the difference

of the initial and final position should be equal to the delta account that has been requested, but in practice it turned out that it was not so.

The sign B of the Encoder had an error of three accounts but, as discussed above, if a count has sent with value W (where W represent represents any value) it moved W but three more counts. That situation is due to a DC motor has a relative Encoder, therefore, the voltage that feeds the motor continues to generate a small delay which causes that other accounts have been produced.

The DC motor voltage to go up resulted different voltage to go down because when the camera went down the gravity force help to required less power than when the camera went up where it has been required more power to overcome the gravity and friction force that have generated.

To finish the section, it can be noticed that the integration of the hardware and the software has been vital to achieve the camera position control with the desired position sent from Matlab software.

The control is called image control, because from the calculation of the focus measures (FM) performed by the autofocus techniques is defined where the best sharpness proceeding to manage to stop the camera movement, that description is part of the macroalgorithm involving the Autofocus.

5.3 Autofocus algorithm results

The autofocus algorithms have been executed and these are the resulting graphics of the focus values which show a graphic for each algorithm containing values for the red, green and blue channels and to see their behaviors.

The focus values have been obtained for a number of 200 frames (images) with the five autofocus techniques in the red, green and blue channels for both Figure 5.5, Figure 5.6 and Figure 5.7.

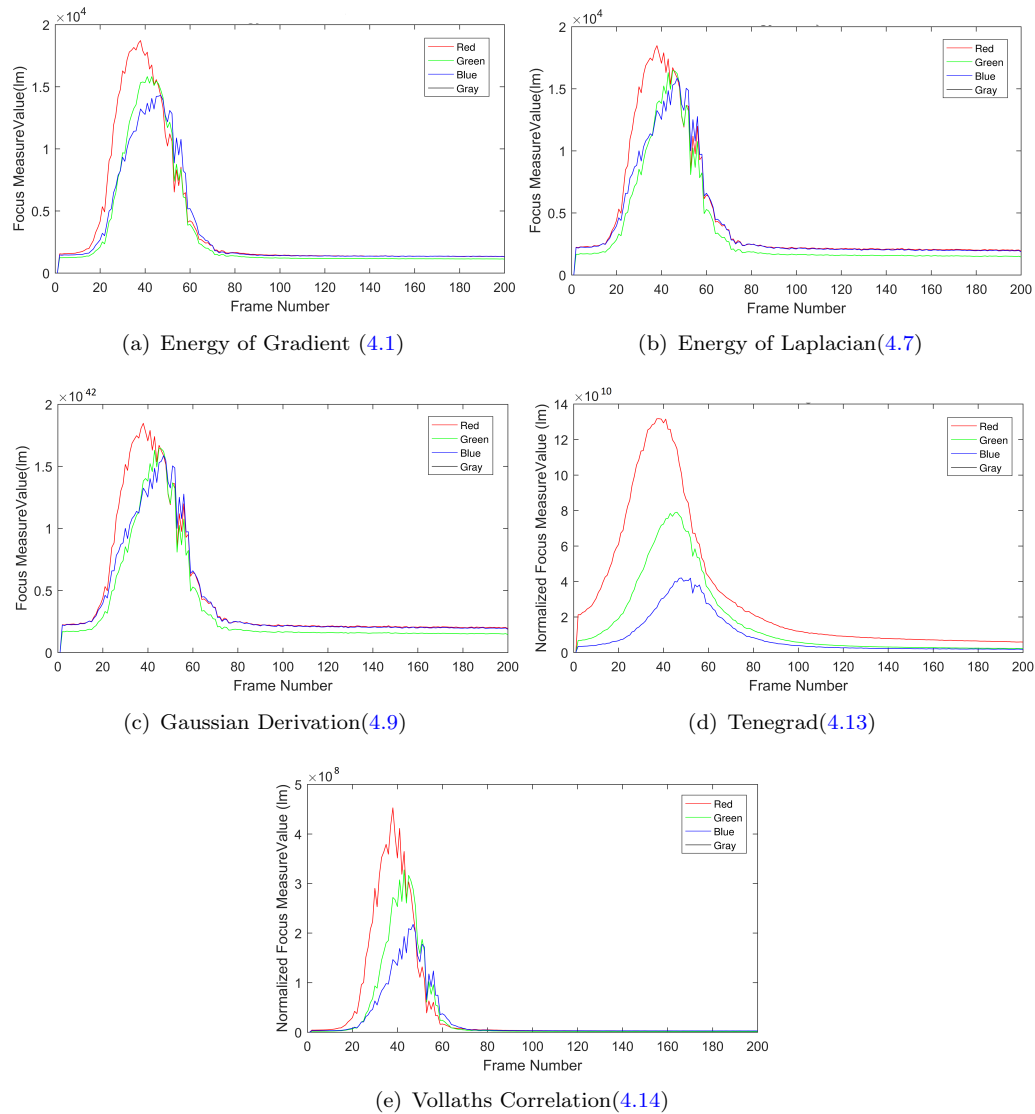


FIGURE 5.5: Focus Measure values graphics represented in red, green and blue channels for the five autofocus techniques with *Sarcophaga sp* when scanning non stop in the best image.

In Figures 5.5 - 5.7 may be noted that most of the curves present noise along its stroke. The reason is because there are external conditions that affect acquire data as they are noise and vibrations of where the goniometer is.

That disturbances are less present in the Tenegrad technique as the graphics denote and several tests have been performed where it has always been consistent that noise and vibrations were quite eliminated with Tenegrad technique. Such a technique involves the convolution of the image with a Sobel filter within their calculations which are quite appropriate for the FM values. The Gaussian Derivation presented the biggest problem with noise and vibrations, followed by Energy of Laplacian and Vollath's Correlation.

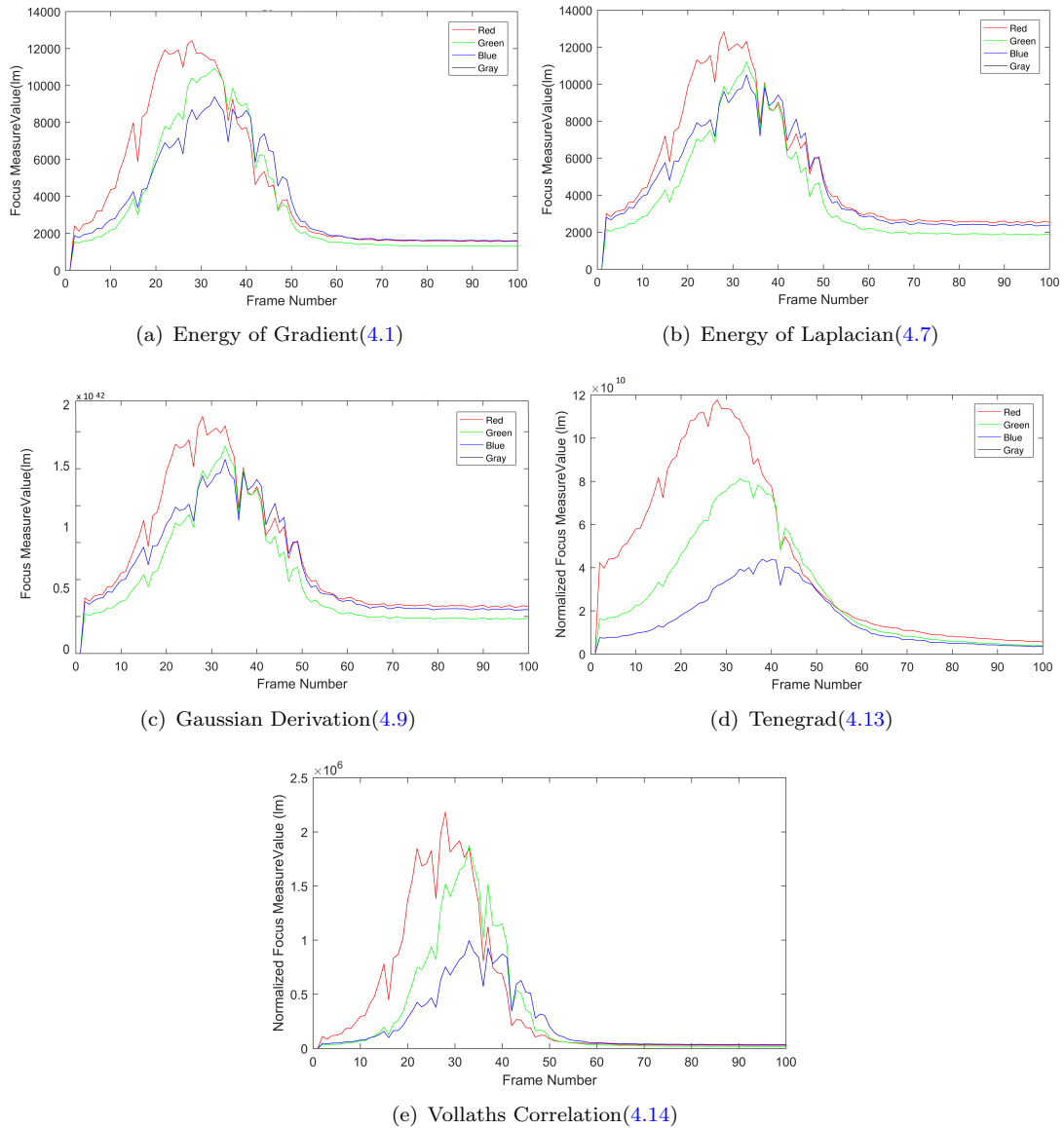


FIGURE 5.6: Focus Measure values graphics represented in red, green and blue channels for the five autofocus techniques with *Scathophaga stercoraria sp* when scanning non stop in the best image.

As for the channels, it is distinguished that the red channel is the fastest one to reach a maximum value in comparison with the other channels. Besides, the red channel presented magnitudes of the focus measurements values greater than in the cases of green and blue channels. Moreover, the red channel has a major difference in the Tenegrad technique, more than in the rest of the autofocus techniques.

The order of the values of FM are quite different. The techniques of Energy of Gradiente, Energy of Laplacian and Vollath's Correlation remain in the order of the power to the eight, the Gaussian Derivation technique works in the order of the power of 42 while

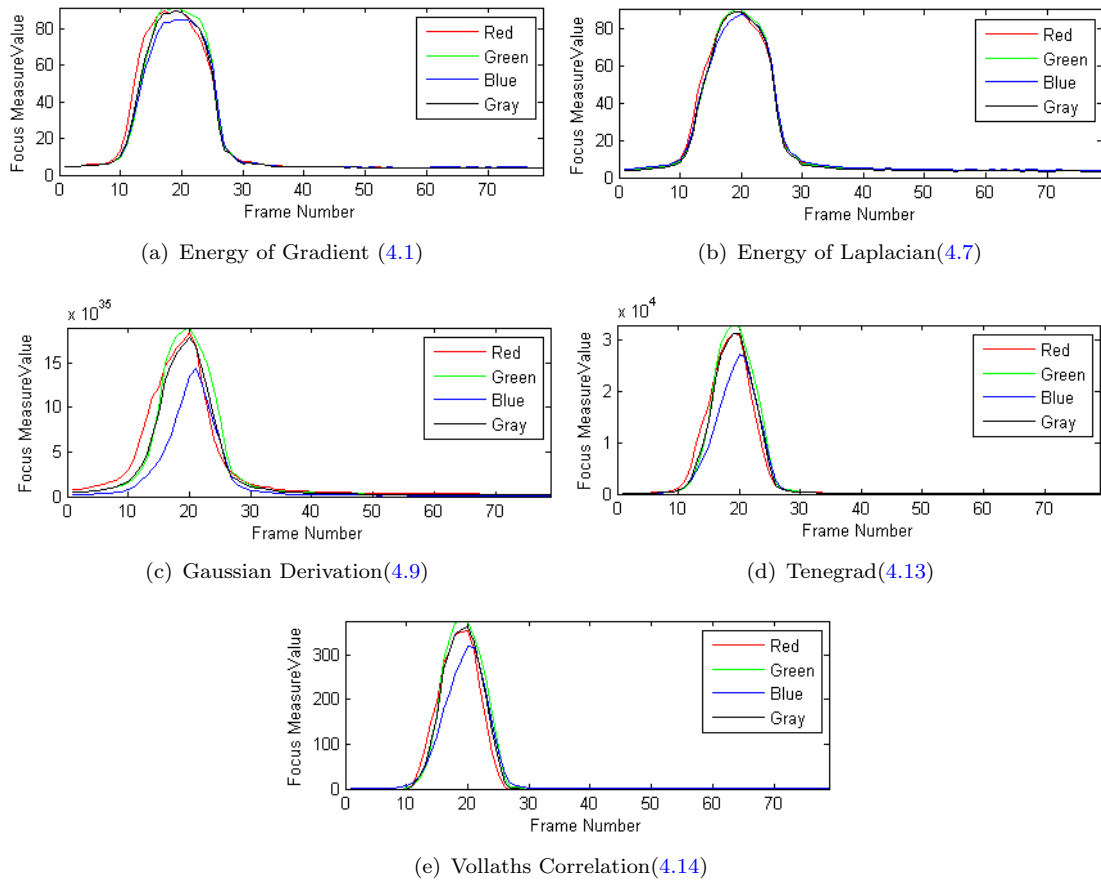


FIGURE 5.7: Focus Measure values graphics represented in red, green and blue channels for the five autofocus techniques with *Calliphora sp* when scanning non stop in the best image.

Tenegrad has exponential values in the order of 10, these results are generated by the mathematics that each carries.

For instance, it can be appreciated the both Figure 5.5, 5.6, and 5.7 result from the mathematical modelling of Chapter 4 (i.e, the corresponding formulae each algorithms are the Energy of Gradient is (4.1), Energy of Laplacian is (4.7), Gaussian Derivation is (4.9), Tenegrad is (4.13) and Vollath's Correlation is (4.14)).

To complement the study of the focus measure values that have been obtained, Tables 5.2 - 5.3 show a summary of the maximum values regarding the technique and the channel.

TABLE 5.1: Summary of the maximum FM values obtained by each algorithm for autofocusing with *Sarcophaga sp.*

Technique	Red Channel		Green Channel		Blue Channel	
	Maximum FM value	Position	Maximum FM value	Position	Maximum FM value	Position
Energy of Gradient	1.87x10 ⁸	37	1.58x10 ⁸	40	1.43x10 ⁷	46
Energy of Laplacian	1.85x10 ⁸	37	1.65x10 ⁸	44	1.59x10 ⁸	46
Gaussian Derivation	1.32x10 ⁴²	37	7.90x10 ⁴¹	45	4.20x10 ⁴¹	47
Tenengrad	4.53x10 ¹⁰	37	3.27x10 ¹⁰	42	2.18x10 ⁹	46
Vollaths Correlation	6.11x10 ⁸	37	4.55x10 ⁸	42	2.94x10 ⁸	46

TABLE 5.2: Summary of the maximum FM values obtained by each algorithm for autofocusing with *Scathophaga stercoralia sp.*

Technique	Red Channel		Green Channel		Blue Channel	
	Maximum FM value	Position	Maximum FM value	Position	Maximum FM value	Position
Energy of Gradient	1.24x10 ⁸	27	1.09x10 ⁸	32	9.39x10 ⁷	32
Energy of Laplacian	1.28x10 ⁸	27	1.12x10 ⁸	32	1.05x10 ⁸	32
Gaussian Derivation	1.18x10 ⁴²	27	8.12x10 ⁴¹	32	4.39x10 ⁴¹	37
Tenengrad	2.18x10 ¹⁰	27	1.88x10 ¹⁰	32	9.96x10 ⁹	32
Vollaths Correlation	2.67x10 ⁸	27	2.34x10 ⁸	32	1.24x10 ⁸	32

TABLE 5.3: Summary of the maximum FM values obtained by each algorithm for autofocusing with *Calliphora sp.*

Technique	Red Channel		Green Channel		Blue Channel	
	Maximum FM value	Position	Maximum FM value	Position	Maximum FM value	Position
Energy of Gradient	2.72x10 ⁸	16	2.67x10 ⁸	23	2.47x10 ⁸	22
Energy of Laplacian	1.96x10 ⁸	16	1.82x10 ⁸	19	1.79x10 ⁸	22
Gaussian Derivation	1.89x10 ³⁷	16	1.72x10 ³⁷	20	1.43x10 ³⁷	22
Tenengrad	3.12x10 ⁹	16	2.79x10 ⁹	18	2.49x10 ⁹	21
Vollaths Correlation	3.87x10 ⁸	16	3.73x10 ⁸	20	3.13x10 ⁸	21

In the last Tables 5.1 - 5.3, the magnitudes of the values in the red channel are greater than in the green and blue channel, due to the photoreceptor pigments of insect *ommata* it captures the red spectrum better than the other channels. The characteristic peculiarity depends on the species, not on all species is given, but in the species used has been evidenced the characteristic.

With the focus measure (FM) data from each of the channels, a comparison is needed to find how large a magnitude has been over what results in Tables 5.4 - 5.6.

TABLE 5.4: Differences between FM magnitudes are shown in percentages to note the difference between channels in *Sarcophaga*.

Technique	Percentages of variation between channels		
	Red-Green(%)	Red-Blue(%)	Green-Blue(%)
Energy of Gradient	15.50	23.56	9.53
Energy of Laplacian	10.85	14.13	3.68
Gaussian Derivation	40.00	68.14	46.90
Tenengrad	27.67	51.92	33.52
Vollaths	25.57	51.90	35.38
Correlation			

TABLE 5.5: Differences between FM magnitudes are shown in percentages to note the difference between channels in *Schathophaga stercoraria*.

Technique	Percentages of variation between channels		
	Red-Green(%)	Red-Blue(%)	Green-Blue(%)
Energy of Gradient	12.11	24.44	14.04
Energy of Laplacian	12.50	18.15	6.46
Gaussian Derivation	30.98	62.69	45.94
Tenengrad	14.09	54.37	46.88
Vollaths	12.65	53.69	46.98
Correlation			

The percentages of variation have been made by means of three comparisons as shown in Tables 5.4 - 5.6 where the greatest difference is evidenced between the red and blue channel. The percentage indicates that red has larger magnitudes with both the green and blue channels. Meanwhile, the green channel has more magnitude than blue, stating once again that the blue channel provides little information regardless of the species of the fly.

TABLE 5.6: Differences between FM magnitudes are shown in percentages to note the difference between channels in *Calliphora*.

Technique	Percentages of variation between channels		
	Red-Green(%)	Red-Blue(%)	Green-Blue(%)
Energy of Gradient	3.69	9.18	5.69
Energy of Laplacian	7.65	8.67	1.10
Gaussian Derivation	8.99	24.34	16.86
Tenengrad	10.58	20.19	10.75
Vollaths	3.61	18.86	15.82
Correlation			

Also, it is said that the images in the red channel contain more information than in the green channel, and there is even less information in the blue channel. Because the images in the blue channel are darker, it improves the clarity in the green and it is better to see images in the red channel. Although, some species is adequate the green channel, in the insects used in the experiment the red channel has always been the the appropriate.

Regarding the position, the position is always the same in each autofocus technique in the red channel, that is to say, it keeps the place where an appropriate focus has been found. The position is not always the same for each fly. In addition, the site has some variations in the cases of the green and blue channels which are not suitable to be used in the algorithm where the camera movement has to be stopped.

In addition, it should be noted that there has been concordance in the results even though it has been worked with different species. The order of power of the FM values of *Sarcophaga* sp. And *Scathophaga stercoraria* are the same as those of *Calliphora* that have been slightly different in terms of Gaussian Derivation . This was claimed to have been produced by environmental conditions such as light, disturbances mentioned above as vibrations when moving or light absorption in the fly occurred differently.

5.3.1 Normalized Algorithms

All the results of the normalized focus values must be in order to be able to compare between techniques. Normalization consists of taking the highest focus measure (FM)

obtained and divide it with the other FM values to be able to have results based on a maximum value of one, as in Figure 5.8- 5.10. Normalized graphs show that the channels overlap in their curves, but that is because they are all based on one; it also preserves the disturbances.

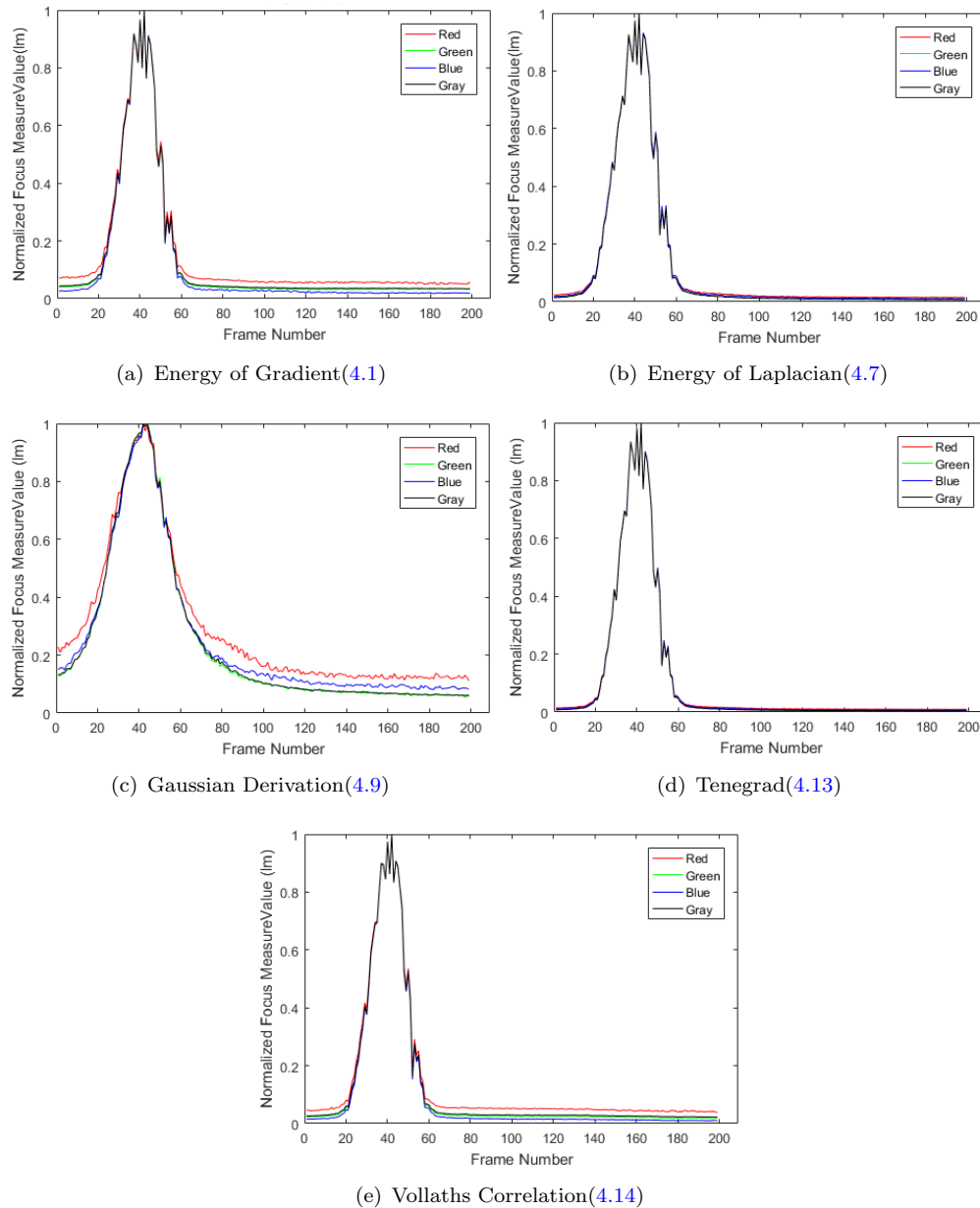


FIGURE 5.8: Focus Measure Normalized values graphics represented in red, green and blue channels for the five autofocus techniques with *Sarcophaga sp* when scanning non stop in the best image.

Now the comparisons between the methods is easier to conduct because there is already a standard. It can be observed that the maximum values are always reached in the range from 20 to 40 frames in *Scathophaga stercoraria sp* and 30 to 50 in *Sarcophaga sp* fly,

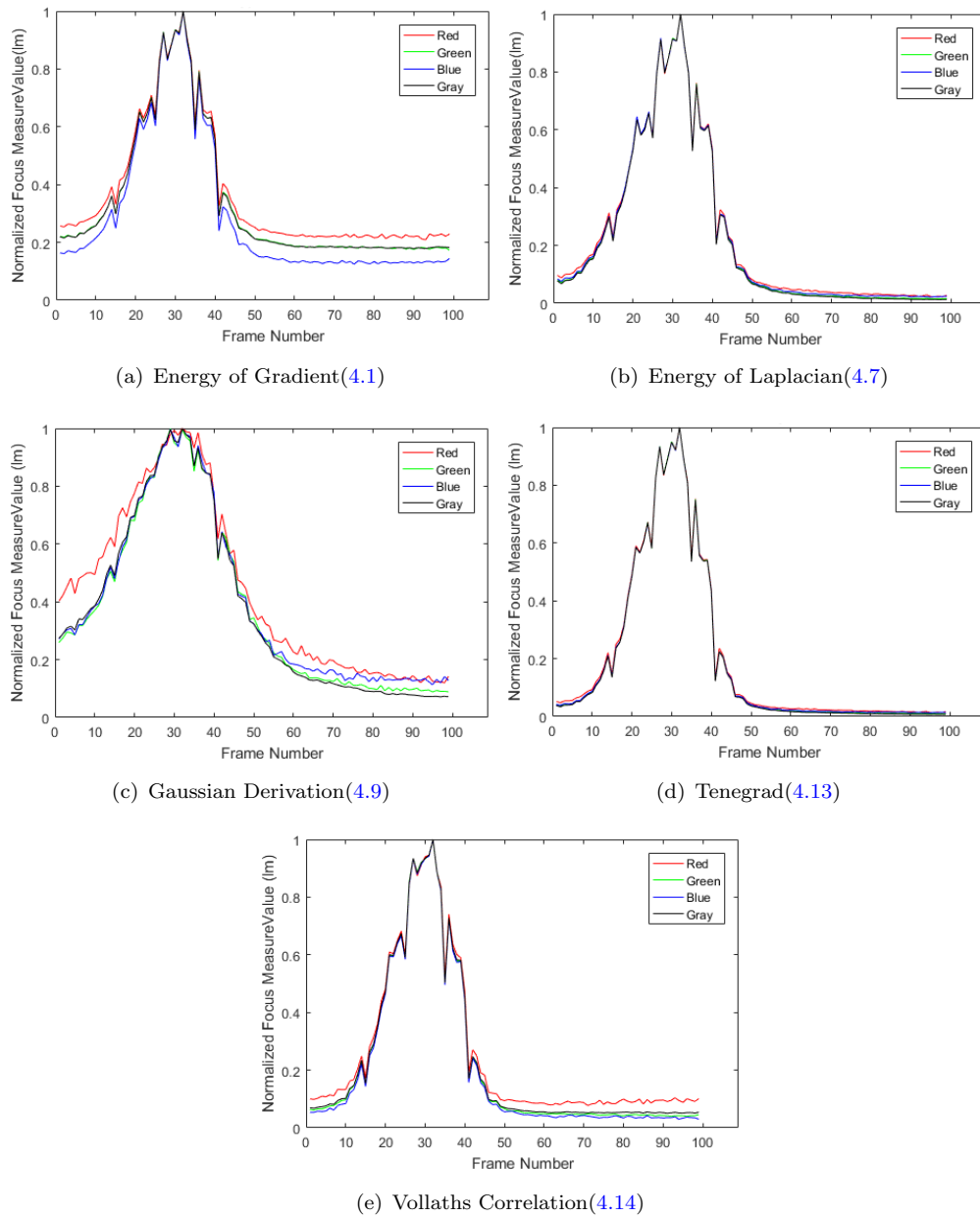


FIGURE 5.9: Focus Measure Normalized values graphics represented in red, green and blue channels for the five autofocus techniques with *Scathophaga stercoraria sp* when scanning non stop in the best image.

so the other values are to notice a gradient of variation between what is clear and what is not. The values never start exactly at zero because when the motor makes its first move it receives a change from the resting state to a motion state generating a little disturbance at the beginning.

With all focus measures that have obtained, offline tests can be done to extract the best image by each of the 5 autofocus techniques in red, green, blue and gray channel. Additionally, the channels, to be normalized as in most of the graphs of Figure 5.8 - 5.10,

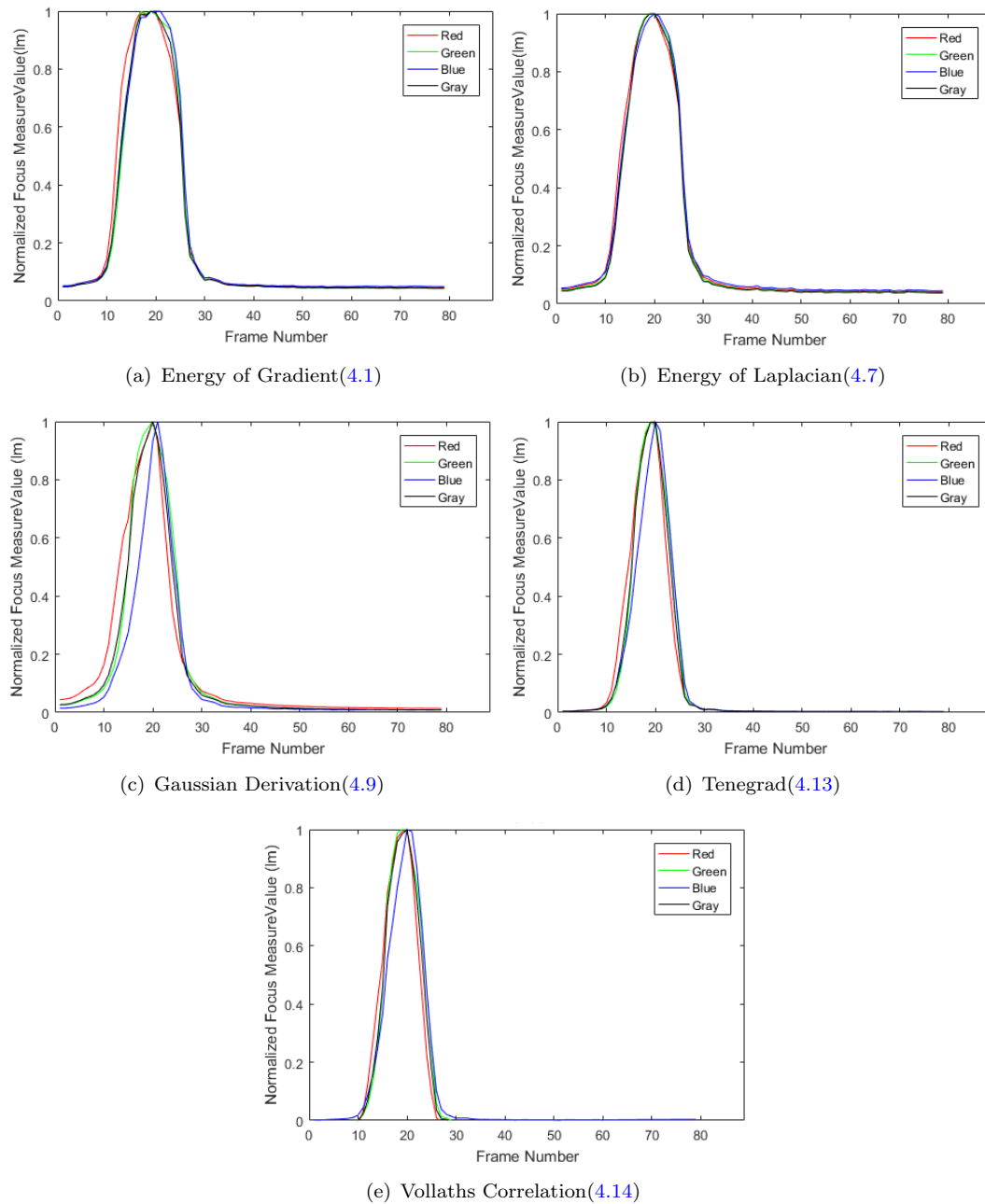


FIGURE 5.10: Focus Measure Normalized values graphics represented in red, green and blue channels for the five autofocus techniques with *Calliphora sp* when scanning non stop in the best image.

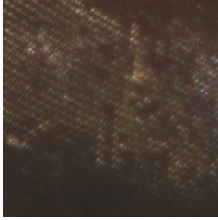
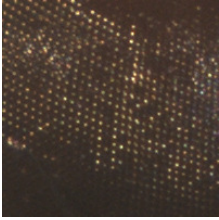
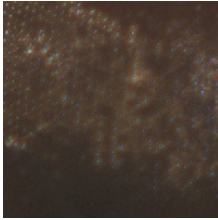
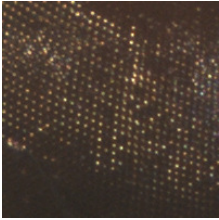
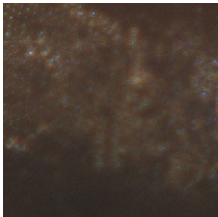
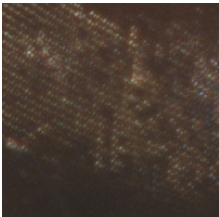
they overlapped because of a very similar behavior between them and Also, the channels to be normalized most of the graphs of Figures 5.8, 5.9 and 5.10 overlapped because it has a very similar behavior between them, indicating that the technique is acceptable. Despite this, the Gaussian technique does not present the last behavior since it is noted that the red channel has higher values and the stroke is separated from the other strokes of the channels.

Moreover, the normalization has helped to visualize that the difference between the unfocused and focused values has on average a gradient of 0.72 for *Sarcophaga*, 0.63 for *Scathophaga* and 0.73 for *Calliphora*. Tenegrad has been the one that presents a much greater gradient in all the cases, that is to say, it gives a reference that provides a distinction between the clear images and images that are not focus.

Having learned that the autofocus techniques that had the best results have been Tenegrad and the Energy of Gradient, Tenegrad in the red channel has chosen to perform the camera positioning. That is to say, as discussed in Section 5.2, with the *AutoSlopes* function it is possible to position the camera in a place where it had a good focus of the image and better observe the compound eye of the insect. Table 5.7, Table 5.8 and Table 5.9 display the focused and unfocused images. Tables 5.7, 5.8 and 5.9 show when the camera stopped and it was in a good place to continue the deep analysis of the eye. Although the other techniques generated results, they have not been relevant for futures studies.

The images have been a product of the combination of all channels to show a complete image where it is important to see the focus degree. The focus is to analyze the internal structure of the eye, for this purpose, in future research should consider that so much depth is required to improve the zoom of the camera and work together with the lighting system (mentioned in Section 2.3.2).

TABLE 5.7: Out of focus and focus images resulting from autofocusing techniques of Energy Gradient, Tenengrad (the two best techniques) and Gaussian Derivation (the technique with most unfavorable results) with a 200x200 zoom of the *Scathophaga stercoraria* sp eye.

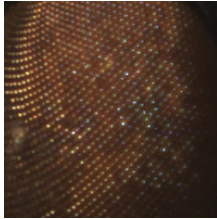
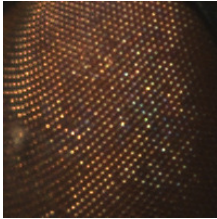
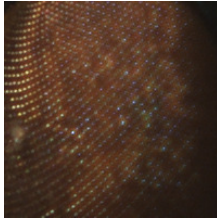
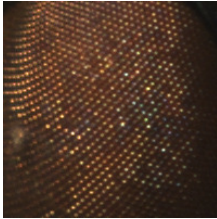
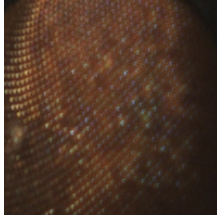
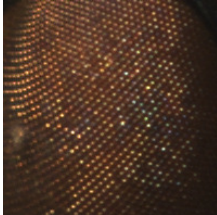
Technique	Out of Focus	In Focus
Energy of Gradient		
Tenengrad		
Gaussian Derivation		

5.4 Data extraction results

Off-line tests have been performed with the data obtained to check if the image with the best sharpness indicated by the algorithms is the appropriate. The implemented includes three types of edge detectors, since it was noticed that the techniques try to detect edges themselves. These types helped understand how well the techniques are when looking for the focus measure.

The tests have been used as an experimental criterion for the detection of the eyes edges, the threshold values have been established empirically until the edges can be noticed. When the first test has achieved, it converted the image to binary image, based on

TABLE 5.8: Out of focus and focus images resulting from autofocusing techniques of Energy Gradient, Tenengrad (the two best techniques) and Gaussian Derivation (the technique with most unfavorable results) with a 200x200 zoom of the *Sarcophaga* eye.

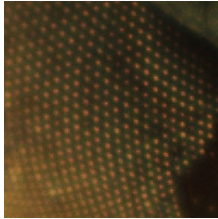
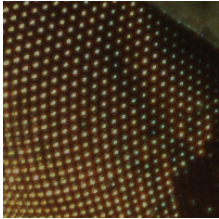
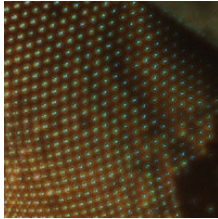
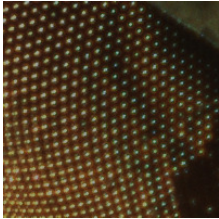
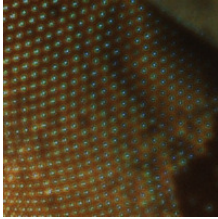
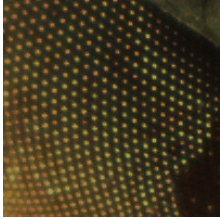
Technique	Out of Focus	In Focus
Energy of Gradient		
Tenengrad		
Gaussian Derivation		

threshold. That is possible using a proper function of Matlab which is based on using a threshold with binarized images. The threshold chosen as $T = 0.2$, which it best detected the edges Of the eyes of the flies as shown in Figure 5.11.

The second test of edge detection consisted in applying a Sobel and Canny filter to the images. The filters used in image processing that are based on searching the gradient of the function of intensity of the image to detect the edges of the figures that make up the image.

Here there are thresholds that try to have a high or low sensitivity to detect an edge. For the Canny filters a threshold of $T = 0.4$ has been used, where if the threshold is higher than the defined threshold, the detection of the edge is on very thick or remarked

TABLE 5.9: Out of focus and focus images resulting from autofocusing techniques of Energy Gradient, Tenengrad (the two best techniques) and Gaussian Derivation (the technique with most unfavorable results) with a 200x200 zoom of the *Calliphora* eye.

Technique	Out of Focus	In Focus
Energy of Gradient		
Tenengrad		
Gaussian Derivation		

lines. If the values have been lower than that established, there is much sensitivity to have more edge than desired.

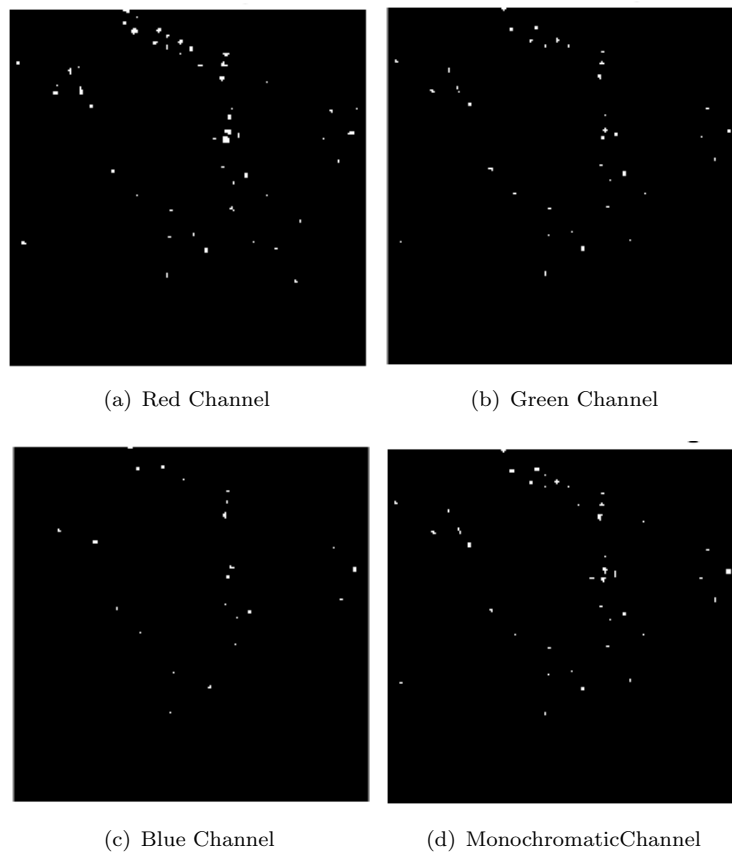


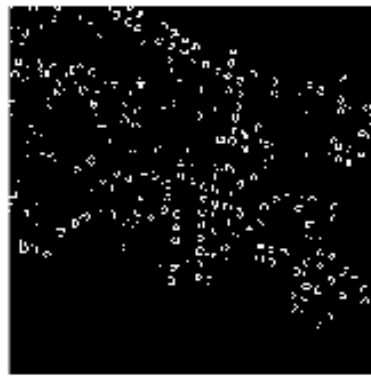
FIGURE 5.11: Edge Detector 1 with a Threshold $t=0.2$, image obtained with the *Scathophaga stercoraria sp* with the Tenegrad autofocus technique in red,green,blue and monocromatic channels.

While the Sobel filter was not able to define a threshold since without threshold the edges have found properly. The tests have been established by setting thresholds of $T = 0.2, 0.4 \text{ or } 0.6$ but their response to the comparison generated by the images without threshold are not very obvious. Figure 5.12 shows the results of second test.

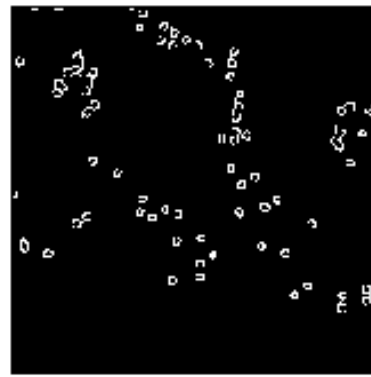
The third test consists of an edge detector that not only detects edges but improves them even when there are disturbances, such as noise and low contrast in the captured image. The algorithm has been created by some authors who sought to improve edge detection under poor conditions of image quality [28].

Poor conditions mean noisy or others disturbance in the image that avoid to identify edges. Third detector only works with monochrome channel images, as shown in Figure 5.13 where the circular blue borders are only visible in the lower right image.

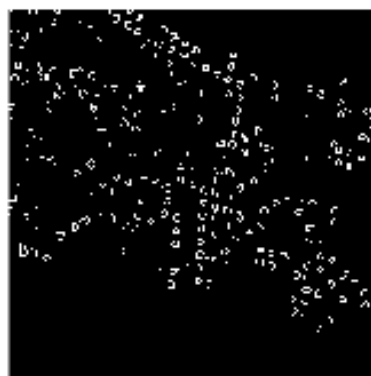
The previous three tests of edge detection have been done with Tenegrad autofocus technique being the best technique since it has a more stable response to disturbances.



(a) Monochromatic Channel SobelFilter



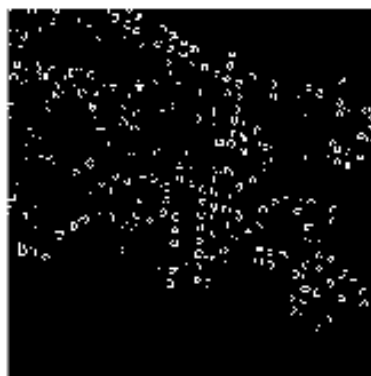
(b) MonochromaticChannel CannyFilter



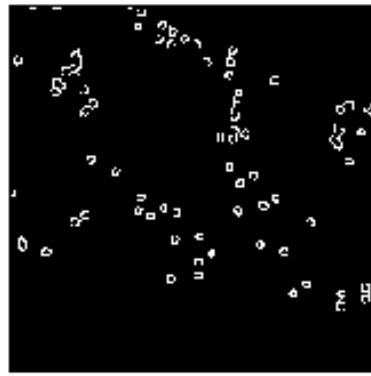
(c) Red Channel SobelFilter



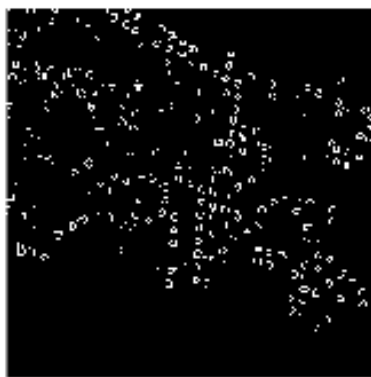
(d) Red Channel CannyFilter



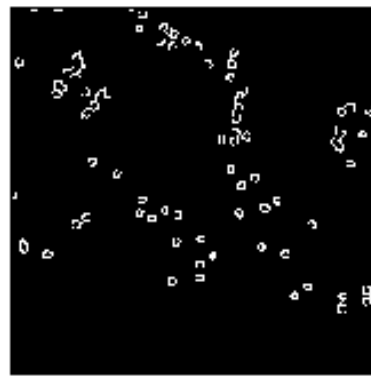
(e) Green Channel SobelFilter



(f) Green Channel CannyFilter



(g) Blue Channel SobelFilter



(h) Blue Channel CannyFilter

FIGURE 5.12: Edge Detector 2 with a Threshold $T=0.4$ with the Canny filter and Sobel filter without threshold, image obtained with the *Scathophaga stercoraria* sp with the Tenegrad autofocus technique in red,green,blue and monochromatic channels.

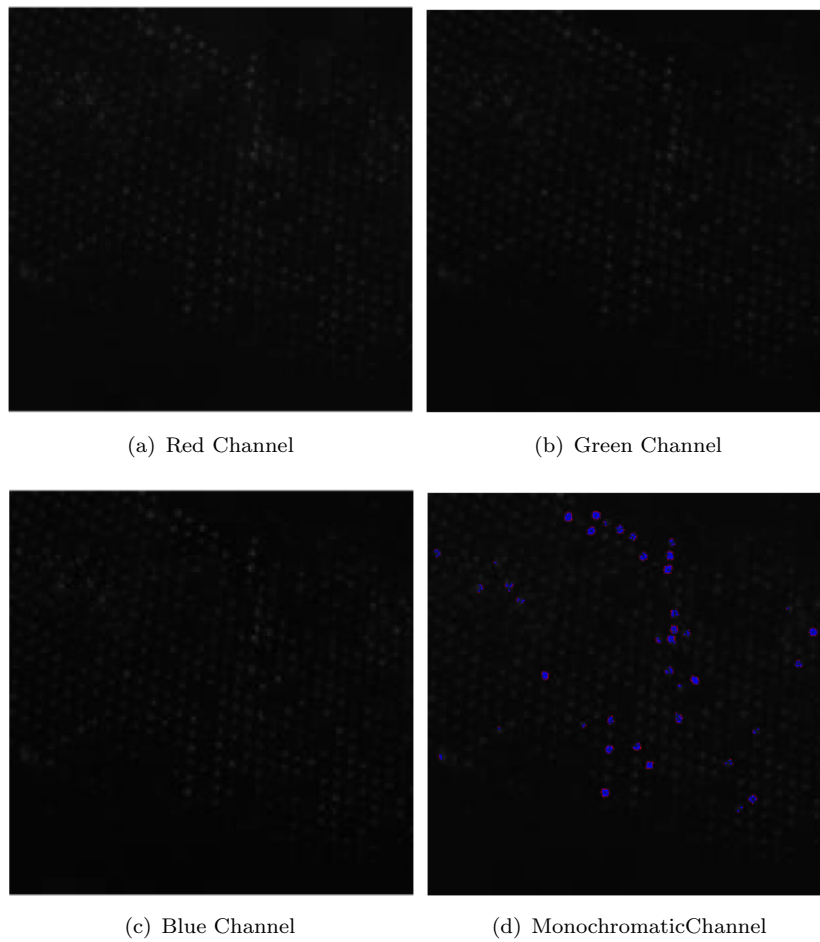


FIGURE 5.13: Edge Detector 3 with the Subpixel Edge detector Filter, image obtained with the *Scathophaga stercoraria sp* with the Tenegrad autofocus technique in red, green, blue and monochromatic channels.

Chapter 6

Discussion

The autofocus algorithm has been proven to obtain sharp images during scanning of the compound eye of an insect through a block-segmented autofocus algorithm (Fig.3.1). It was possible to appreciate the interaction of the micro-algorithm with the macro-algorithm which are part of the autofocus algorithm and it has an adequate operation of image captures with the GRACE second version device.

The live video (Fig.5.1) is being carried out without stopping or the image has frozen, the video streaming is about 19 FPS. When executing the macroalgorithm to find the best approach with a continuous video this helps to capture suitable images in a 200x200 size as the camera moves around the eye. Take into account the image size was useful to apply the autofocus techniques. The images require a lighting system which provides a better quality in the images, because in GRACE second version, the lighting consisted only of two lamps directing their light towards the insect.

As expected, the serial communication between the Arduino, the camera and the software in Matlab has been successful interaction between themselves, the synchronization of the serial parameters have to be the same when configured to ensure communication.

The values of the displacement counts to move the DC motor through the serial communication, has been achieved by the interruptions, it means, the system attended to the motion request as soon as the signals were generated, a system that immediately attends it which is requested. If there is a problem in the communication, in the command line of Matlab, the message shown in 5.1, goes out several times to achieve the communication, and if it is not achieved the system stops and the program must be re-initialized.

The count values of the camera displacement have an error value of the three magnitudes. It has been accepted since the moment the camera stops in the calculated position, at the well focused image. Also, the error is related to the DC motor which contains a relative Encoder which barely stops receiving the voltage for its movement has to be stopped, but the internal signals of the encoder continue sending a bit later (the 3 more values) until the motor voltage is zero.

The *AutoFocusGrace* function (Fig.5.3) has adequately performance for the calculation of the five autofocus techniques. In spite of this, its way of stopping the movement has not been very useful since it is based on the Encoder counts which have an error that tend to not being able to reach the right position.

The *AutoSlopes* function has been implemented to position the camera through image processing, it is an image-based control where it has been conceived make use of the best autofocus technique in only one of its channels. The technique chosen was Tenengrad in the red channel, the data set was divided into groups of five with which the slope was calculated.

The slope method has been able to find a place by identifying abrupt changes of the data and together with the established conditions has been achieved the behavior necessary to stop the movement of the camera. Despite this, the positioning when executed had several problems, since the non-linearity of the system had not been contemplated, which includes the gravity and the friction of the system when moving. The above problems affected the function but it is still functional because the movement is continuous.

The functionality of the *AutoSlopes* algorithm has been properly integrated with microalgorithm of the goniometer both to obtain focus measures with the five techniques as well as to achieve a place where the camera is able to capture images.

The implementation of the *Focus* structure to store the focus measure and *Scanned* to store the captured images has had an important utility because it is required to have the complete data per channel for each focus technique to perform tests outside the execution of the algorithm. Since there is data analysis that can not be performed in the same test, such as finding the differences of the focus measure between channels.

The validation of the autofocus techniques have been carried out by the execution of the algorithms corresponding to their mathematics as Energy of Gradient (4.1), Energy of

Laplacian (4.7), Gaussian Derivation(4.9), Tenengrad (4.13) and Vollath's Correlation (4.14).

Knowledge of the nature of the mathematical basis in the autofocus techniques have explained that these are based on the search on edge detection and the focus measure (FM). Focus measures give an average of the intensity of the image pixels to know the extent of their quality.

The implementation included that the five autofocus techniques were carried out at the same time, the results of the algorithms (Fig.5.5 - 5.7) were advantageous to show differences between them.

The channel analysis helped to know that the red channel generates higher and faster values than other channels, regardless of the type of fly that has been used. As Tables 5.4 - 5.6 has shown there are amplitudes differences between the channels, the difference between the red and blue channels is greater than with the red and green channels.

The Tenengrad technique has proven to be the best for its resistance to noise and vibration disturbances and to have a greater difference between channels; for instance, between the red and green channel there is a difference of 27.68% and between the red and blue channel a difference of 51.91% with the *Sarcophaga* fly.

It has also been noticed that the techniques indicate a great contrast between what is focused and out of focus for which an average gradient of 0.72 was obtained for *Sarcophaga*, 0.63 *Scathophaga stercoraria* and 0.73 for *Calliphora*.

Edge detection techniques have been used to find composite eye edges where threshold values were selected by experimental criteria until the desired shape has been detected. The first technique consisted of a binarization that generated borders in all channels, highlighting the region of the eyes. The Canny and Sobel filters, second technique, are very used in the image processing, so it proved and had an adequate response in all the channels. The third test proved to be accepted only in the monochromatic channel because only grayscale images accept the technique which generates small arrows around the edges. The three ways of edge detection have been used to just ensure the best product image where the camera stopped.

The autofocus algorithm implementation has produced a satisfactory result for the GRACE second version device. It has been possible to verify which was the best technique and channel as well as to use that information to make a new way to control the camera positioning.

Chapter 7

Conclusion and Recommendations

7.1 Conclusions

The implementation of the autofocus algorithm has been achieved through the structuring in five modules (Section 3.3.1) to divide the procedures and have an appropriate response by relating a control circuit (hardware) to the software programs.

Bringing the integration of software and hardware by making possible the unification between the macro-algorithm and the micro-algorithm that set up the implementation of the autofocus algorithm.

The search of the mathematics of each of the five algorithms served to validate them because they are algorithms that make calculations based on the detection of edges. The results of these calculations have been the focus measure, as it was denominated in the project, measured in lumens and that generated an average of the intensity of the pixels to look for the sharpness of the images.

With the execution of the algorithms, it has been possible to find out what is the technique with quick results and precise time to find sharpest images and added to the functional prototype to GRACE second version.

The design is capable of meeting the requirements and restrictions requested in Chapter 3 and 4. The algorithm by modules achieved an organized methodology to position the

camera as well as to unify in a single execution the analysis of the five techniques in its three channels at the same time to be stored in a data structure.

By mean of focus measure, the *AutoSlope* implementation is made when discovering that the grouping of five of these measures is functional a process of stopping the algorithm when finding images with high magnitudes of FM, as it shows in Tables 5.1-5.3.

The implementation threw results represented in graphs and tables, which led to the discovery that the best algorithm was Tenegrad by resistance to perturbations. In addition, the red channel in all cases has had measurements of greater magnitude and achieved some frames before the other channels. Although the results were obtained with different flies, they have been consistent in matching the best technique and channel.

In addition, it included the communication without loss of data and capture uninterrupted images when making movement, as it has been showed in Chapter 5.

The autofocus algorithm evidenced to be functional to perform the following processes:

- The autofocus algorithm control has been accomplished by means of image-based control, using five images grouped to calculate the slope which manage to stop the displacement of the camera that acquires images.
- Storage of focus measure data in the *Focus* structure, for offline analysis such as the calculation of maximum measures, difference gradient between focus and out of focus images, and image edge detection.
- Focus measure calculation on lumens units in different orders due to the autofocus techniques in each of the channels.
- The acquired image are stored in a structure called *Scanned*, with a format of $(X, Y, channel, frame number)$.
- Generation of graphs of focus measure versus number of frames by each of the techniques with their respective channels.
- Image capture in size of 200x200 pixels.
- Serial communication with immediate attention of data, when implementing an algorithm with interruptions attention, process that produced images without frozen effect.

The use of a better positioning control and the tests that unite the whole system of the goniometer GRACE makes the site open to receive modifications to improve for future investigations. The possible control would include conditions of friction and gravity to counteract some difficulties when moving the position of the camera up or down.

7.2 Recommendations (Outlook)

The use of a better positioning control and the tests that unite the whole system of the goniometer GRACE makes possible to receive modifications to improve for future investigations. The new control would include conditions of friction and gravity to counteract some difficulties when moving the position of the camera up or down.

In order to capture images, a good lighting system is recommended, as well as to perform the eye scanning on a surface where aspects such as noise or vibrations are not perceived so much.

For user use is more interactive, it is recommended to do an graphic interface where the user chose the techniques, the amount of accounts to move and next to that include the visualization of the live images that are currently seen. As well as a manual control which manages to put the camera without using the connection with the software, just as the researchers want to do tests without using the techniques of autofocus.

The electronic circuit has to integrate their all in a PCB, currently each integrated components are separately and connected by means of cables indicated by colors, which its aesthetic is crude in sight and difficult to identify for improvements or add some other tool.

Bibliography

- [1] Tamara R Spanier. The Design and Implementation of the Full Actuation for the Five DOF 3D Robotic Scanner. (July), 2015.
- [2] Jesus M. de la Cruz Gonzalez Pajares. *Vision por Computador Imagenes digitales y aplicaciones*. 2002. ISBN 84-7897-472-5.
- [3] D G Stavenga. Pseudopupils of compound eyes. In *Handbook of Sensory Physiology*, volume VII/6A, chapter seven, pages 357–439. Groningen, 1979. ISBN 3-540-08837-7; 0-387-08837-7.
- [4] Vargas-Delgado. An Image Stitching Algorithm for Compound Vision Research. 2016.
- [5] Doekele G Stavenga. Reflections on colourful ommatidia of butterfly eyes. *The Journal of experimental biology*, 205(Pt 8):1077–1085, 2002. ISSN 0022-0949.
- [6] William Retana. Design and implementation of a 6 degrees of freedom control for the 3D robotic scanner at the Groningen University. *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, pages 14–15, 2016.
- [7] Ketan Kumar Sahu. Automated image feature identification for motorised mapping of insect eyes. In *University of Groningen*, 2017.
- [8] Michael F Land and Dan-Eric Nilsson. Animal eyes. *Oxford Animal Biology Series*, 2(1):221, 2002. doi: 10.1093/acprof:oso/9780199581139.001.0001.
- [9] Universidad de Costa Rica. Visin de los insectos. *Museo de Insectos*, 2009. URL <http://www.miucr.ucr.ac.cr/como20ven.html>.

- [10] Andrs Vargas-Delgado. Characterization of the visual space of butterfly eyes via a robotic scanner. *Book of Abstracts 35th Benelux Meeting on Systems and Control*, page 103, 2016. ISSN 978-90-365-4035-3.
- [11] G J Doornbos. Design and implementation of an autofocusing algorithm for a 3D robotic scanner IEM Bachelor thesis. (June), 2015.
- [12] John K. Douglass and Martin F. Wehling. Rapid mapping of compound eye visual sampling parameters with FACETS, a highly automated wide-field goniometer. *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, pages 1–13, 2016. doi: M.4.838-2001.
- [13] Arturo de la Escalera Hueso. Vision por computador:fundamentos y metodos. pages 1–63, 2001. doi: 10.1007/s00359-016-1119-7.
- [14] Gonzalo Pajares and JesusM. de la Cruz. *Visin por Computador:Imágenes Digitales y Aplicaciones*. 2002. ISBN 84-7897-472-5.
- [15] Wendy B. Rauch-Hindin. *Aplicaciones de la inteligencia artificial en la actividad*. 1989. ISBN 8487189075. URL <https://books.google.com/books?isbn=8487189075>.
- [16] Rafael C Gonzalez and Richard E Woods. *Digital Image Processing (3rd Edition)*. 2007. ISBN 013168728X. URL <http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20{&}path=ASIN/013168728X>.
- [17] Fernando Jordn Monts. La visin de los insectos. *Investigacion y Ciencia*, Septiembre,2013. URL <http://www.investigacionyciencia.es/revistas/investigacion-y-ciencia/numero/444/la-visin-de-los-insectos-11363>.
- [18] Angulo Garcia, Angulo. *Sistemas Digitales y tecnologia de computadores*. 2007. URL <https://books.google.nl/books?id=i8eX0aMzmzcC&lpg=PA147&dq=que%20es%20cmos&hl=es&pg=PR2#v=onepage&q=que%20es%20cmos&f=false>.
- [19] Diego Gonzalez-Aguilera, Javier Gomez-Lahoz, and Pablo Rodríguez-Gonzálvez. An automatic approach for radial lens distortion correction from a single image. *IEEE Sensors journal*, 11(4):956–965, 2011.

- [20] J M Geusebroek, F Cornelissen, a W Smeulders, and H Geerts. Robust autofocusing in microscopy. *Cytometry*, 39(1):1–9, 2000. ISSN 0196-4763. doi: 10.1002/(SICI)1097-0320(20000101)39:1<1::AID-CYTO2>3.0.CO;2-J.
- [21] Eric Krotkov. Focusing. *International Journal of Computer Vision*, 1(3):223–237, 1988. ISSN 09205691. doi: 10.1007/BF00127822.
- [22] Dieter Vollath. Automatic focusing by correlative methods. *Journal of Microscopy*, 147(3):279–288, 1987. ISSN 00222720. doi: 10.1111/j.1365-2818.1987.tb02839.x. URL <http://doi.wiley.com/10.1111/j.1365-2818.1987.tb02839.x>.
- [23] Accommodation in Computer Vision. Accommodation in computer vision. 1970.
- [24] Pittsburgh Pr, Robotics Inst, and J F Schlag El. IMPLEMENTRTION OF AUTOMATIC FOCUSING ALGORITHMS FOR A mohEEEmhmiI EhEmohEo-hEohmI EEEEmhEEEEEmhhE mE ~ hEEEE ~ IEE. 1983.
- [25] TTE Yeo, SH Ong, Jayasooriah, and R. Sinniah. Autofocusing for tissue microscopy. *Image and Vision Computing*, 11(10):629–639, 1993. ISSN 02628856. doi: 10.1016/0262-8856(93)90059-P.
- [26] Bernd Jahne. *Digital Image Processing (3rd Edition)*. 1995. ISBN 013168728X.
- [27] D. Vollath. The influence of the scene parameters and of noise on the behaviour of automatic focusing algorithms. *Journal of Microscopy*, 151(2):133–146, 1988. ISSN 13652818. doi: 10.1111/j.1365-2818.1988.tb04620.x.
- [28] Aleman-Flores Santana-Cedres Trujillo-Pino, Krissian. Accurate subpixel edge location based on partial area effect. (October), 2012.

Appendix A

List Code

The code used by the Arduino appears in the list using pseudocode, this code implements commands from Matlab, interprets them and proceeds to execute its different functions.

List 4.1 Initialize Camera Pseudocode

```
1  vid is a global variable
2  vid = input video from Pointgrey camera;
3  src = assign to src vid video source;
4  Set the shutter mode to automatic
5  Set the exposure mode to automatic
6  Set the gain in automatic
7  Sets up WhiteBalanceRB in [589 901];
8  clear
9  show 'End of section 3' in the command line
```

List 4.2 Process Image Dimension

```
1
2 clear X Y
3 frame = input video;
4 show the frame decrease the intensity with multiplication by 10
5 [X Y] = ginput(1)
6 set the ROI Position with the camera maximum size
7 XOffset = X-100;
8 YOffset = Y-100;
```

```

9 WidthX = 200;
10 HeightY = 200;
11 assign the
12 Call the ROIPosition function and designate the desired size

```

List 4.3 Serial Communication Pseudocode

```

1
2 if exist 'leo' and 'var' variables
3   close leo;
4   delete leo;
5   clear leo;
6 finish condition
7 clear the command window line
8 show in the Command Line the phase 'End of section 6'
9
10 clear the command window
11 define leo a global variable
12 leo = Create serial port object;
13 %set up serial properties
14 set 'DataBits' in leo as 8;
15 set 'StopBits' in leo as 1;
16 set 'BaudRate' in leo as 9600;
17 set 'Parity' in leo as 'none';
18 set 'Timeout' in leo as 60
19
20 set leo with the interrupt callback
21 Open the connection with the Arduino.
22 show the message 'End of section 7' in commad line

```

List 4.4 AutoSlopes Algorithm

```

1 AutofocusSlopes_12_June(vid, leo)
2 Cycles = 200;
3 delta = 10; %Step size of get the photos, It could be the distance send by
   Matlab to Arduino that make to move the position camera and et a
   snapshot to save in the video %
4 Focus = zeros(Cycles, 47);
5 figure
6 ylim([0 20000])

```



```

7   xlim([0 Cycles])
8   grid on
9   hold on
10  SlopeCondition = false;
11  i = 1;
12  DataSet = 5;
13
14 while (i <= Cycles);
15     Focus(i,1) = i; %save the samples number in Focus structure
16     if i==1
17         Focus(i,2)=Cycles/2*delta;
18         TeleMove_10May(leo,Focus(i,2));
19         pause(2);
20     else
21         Focus(i,2)=-delta;
22         TeleMove_10May(leo,Focus(i,2));
23         pause(0.3);
24
25         %im = getsnapshot(vid);
26         im = AverageSnapGRACEOriginal(vid,3);
27         pause(0.1);
28         Scanned(:, :, :, i) = im; %save the im in th Scanned
29         %***** AutoFocus Techniques*****
30         %EnergyofGradient
31         [Focus(i,3)] = EnGraAcTest(im(:, :, 1));
32         [Focus(i,4)] = EnGraAcTest(im(:, :, 2));
33         [Focus(i,5)] = EnGraAcTest(im(:, :, 3));
34         %Energyof Laplacian
35         [Focus(i,9)] = EneLapTest(im(:, :, 1));
36         [Focus(i,10)] = EneLapTest(im(:, :, 2));
37         [Focus(i,11)] = EneLapTest(im(:, :, 3));
38         %Gaussian Derivate
39         [Focus(i,15)] = GauDerTest(im(:, :, 1));
40         [Focus(i,16)] = GauDerTest(im(:, :, 2));
41         [Focus(i,17)] = GauDerTest(im(:, :, 3));
42         %Tenengrad
43         [Focus(i,21)] = TengradTest(im(:, :, 1));
44         [Focus(i,22)] = TengradTest(im(:, :, 2));
45         [Focus(i,23)] = TengradTest(im(:, :, 3));
46         %Vollath 's Corelation
47         [Focus(i,27)] = VollathTest(im(:, :, 1));

```

```

48     [Focus(i,28)]=VollathTest(im(:, :, 2));
49     [Focus(i,29)]=VollathTest(im(:, :, 3));
50
51     if i >= DataSet+3
52         %***** Compute the slope *****
53         %Slope to EnerGrad
54         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,3),1);
55         Focus(i-1,6)=p(1);
56         Focus(i-1,33)=mu(2);
57         clear p mu
58         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,4),1);
59         Focus(i-1,7)=p(1);
60         Focus(i-1,34)=mu(2);
61         clear p mu
62         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,5),1);
63         Focus(i-1,8)=p(1);
64         Focus(i-1,35)=mu(2);
65         clear p mu
66         %Slope to EnerLap
67         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,9),1);
68         Focus(i-1,12)=p(1);
69         Focus(i-1,36)=mu(2);
70         clear p mu
71         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,10),1);
72         Focus(i-1,13)=p(1);
73         Focus(i-1,37)=mu(2);
74         clear p mu
75         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,11),1);
76         Focus(i-1,14)=p(1);
77         Focus(i-1,38)=mu(2);
78         clear p mu
79         %Slopes to GauDer
80         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,15),1);
81         Focus(i-1,18)=p(1);

```

```
82         Focus(i-1,39)=mu(2);
83         clear p mu
84         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,16),1);
85         Focus(i-1,19)=p(1);
86         Focus(i-1,40)=mu(2);
87         clear p mu
88         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,17),1);
89         Focus(i-1,20)=p(1);
90         Focus(i-1,41)=mu(2);
91         clear p mu
92         %Slope to Tengrad
93         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,21),1);
94         Focus(i-1,24)=p(1);
95         Focus(i-1,42)=mu(2);
96         clear p mu
97         [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,22),1);
98         Focus(i-1,25)=p(1);
99         Focus(i-1,43)=mu(2);
100        clear p mu
101        [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,23),1);
102        Focus(i-1,26)=p(1);
103        Focus(i-1,44)=mu(2);
104        clear p mu
105        %Slope to Vollath
106        [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,27),1);
107        Focus(i-1,30)=p(1);
108        Focus(i-1,45)=mu(2);
109        clear p mu
110        [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,28),1);
111        Focus(i-1,31)=p(1);
112        Focus(i-1,46)=mu(2);
113        clear p mu
114        [p,~,mu] = polyfit(Focus(i-DataSet:i-1,1),Focus(i-DataSet:i
-1,29),1);
```

```

115         Focus(i-1,32)=p(1);
116         Focus(i-1,47)=mu(2);
117         clear p mu
118         %*****END Compute Slopes*****
119     else
120         %Blank spaces in case it is not necessary compute the slope
121         Focus(i,6) = 0;
122         Focus(i,7) = 0;
123         Focus(i,8) = 0;
124
125         Focus(i,12) = 0;
126         Focus(i,13) = 0;
127         Focus(i,14) = 0;
128
129         Focus(i,18) = 0;
130         Focus(i,19) = 0;
131         Focus(i,20) = 0;
132
133         Focus(i,24) = 0;
134         Focus(i,25) = 0;
135         Focus(i,26) = 0;
136
137         Focus(i,30) = 0;
138         Focus(i,31) = 0;
139         Focus(i,32) = 0;
140     end
141
142     if (i>=DataSet+3) %Comparison previous and the current slope
143         %Condition
144         ConditionChannel = 24;
145         if (Focus(i-1,ConditionChannel) < Focus(i-2,ConditionChannel))
146             && (sign(Focus(i-1,ConditionChannel)) == -1) %if this condition is true,
147                 Motor must be to stop
148                 SlopeCondition = true;
149                 i%print indicator
150                 break
151         end
152     end
153     figure(2);
154     subplot(3,2,1)

```

```
154     plot(Focus(1:i,1),Focus(1:i,3),'r',Focus(1:i,1),Focus(1:i,4),'g',Focus
(1:i,1),Focus(1:i,5),'b');
155     title('Energy of Gradient');
156     xlabel('Frame Number');
157     ylabel('Focus MeasureValue(lm)'); hold on;
158     %figure(3);
159     subplot(3,2,2)
160     plot(Focus(1:i,1),Focus(1:i,9),'r',Focus(1:i,1),Focus(1:i,10),'g',
Focus(1:i,1),Focus(1:i,11),'b');
161     title('Energy of Laplacian');
162     xlabel('Frame Number');
163     ylabel('Focus MeasureValue(lm)'); hold on;
164     %figure(4);
165     subplot(3,2,3)
166     plot(Focus(1:i,1),Focus(1:i,9),'r',Focus(1:i,1),Focus(1:i,10),'g',
Focus(1:i,1),Focus(1:i,11),'b');
167     title('Gaussian Derivation');
168     xlabel('Frame Number');
169     ylabel('Focus MeasureValue(lm)'); hold on;
170     %figure(5);
171     subplot(3,2,4)
172     plot(Focus(1:i,1),Focus(1:i,15),'r',Focus(1:i,1),Focus(1:i,16),'g',
Focus(1:i,1),Focus(1:i,17),'b');
173     title('Tenengrad');
174     xlabel('Frame Number');
175     ylabel('Normalized Focus MeasureValue (lm)'); hold on;
176     %figure(6);
177     subplot(3,2,5)
178     plot(Focus(1:i,1),Focus(1:i,21),'r',Focus(1:i,1),Focus(1:i,22),'g',
Focus(1:i,1),Focus(1:i,23),'b');
179     title('Vollaths Correlation');
180     xlabel('Frame Number');
181     ylabel('Normalized Focus MeasureValue (lm)'); hold on;
182
183     i = i+1;
184 end
185 end
```

List 4.5 AutoFocusGRACE Algorithm

```

1 AutofocusTestGrace_12_June(vid, leo)
2 Cycles = 100;
3 delta = 15; %Step size of get the photos, It could be the distance send by
   Matlab to Arduino that make to move the position camera and et a
   snapshot to save in the video %
4
5 Focus = zeros(Cycles,8);
6
7 figure
8 ylim([0 10000])
9 xlim([0 Cycles])
10 grid on
11 hold on
12 for i=1:Cycles
13 Focus(i,1) = i; %save the samples number in Focus structure
14 %move Up to Down the camera to capture the image(im)
15 if i==1
16     Focus(i,2)=Cycles/2*delta;
17 else
18     Focus(i,2)=-delta;
19 end
20
21 [EncB ,EncA] = TeleMove_10May(leo ,Focus(i,2));
22 Focus(i,3) = EncB;
23 Focus(i,4) = EncA;
24 Focus(i,5) = EncB-EncA;
25 %im = getsnapshot(vid);
26 im = AverageSnapGRACEOriginal(vid,3);
27 pause(0.1);
28
29 Scanned(:, :, :, i) = im; %save the im in th Scanned structure
30
31 %Add here the autofocus techniques
32 %Energy of Gradient
33 [Focus(i,6)] = EnGraAcTest(im(:, :, 1));
34 pause(0.1);
35 [Focus(i,7)] = EnGraAcTest(im(:, :, 2));
36 pause(0.1);
37 [Focus(i,8)] = EnGraAcTest(im(:, :, 3));
38 %Energy of Laplacian

```

```

39   [Focus(i,9)] = EneLapTest(im(:, :, 1));
40   pause(0.1);
41   [Focus(i,10)] = EneLapTest(im(:, :, 2));
42   pause(0.1);
43   [Focus(i,11)] = EneLapTest(im(:, :, 3));
44   %Gaussian Derivation
45   [Focus(i,12)] = GauDerTest(im(:, :, 1));
46   pause(0.1);
47   [Focus(i,13)] = GauDerTest(im(:, :, 2));
48   pause(0.1);
49   [Focus(i,14)] = GauDerTest(im(:, :, 3));
50   %Tenanrad
51   [Focus(i,15)] = TengradTest(im(:, :, 1));
52   pause(0.1);
53   [Focus(i,16)] = TengradTest(im(:, :, 2));
54   pause(0.1);
55   [Focus(i,17)] = TengradTest(im(:, :, 3));
56   %Vollaths Correlation
57   [Focus(i,18)] = VollathTest(im(:, :, 1));
58   pause(0.1);
59   [Focus(i,19)] = VollathTest(im(:, :, 2));
60   pause(0.1);
61   [Focus(i,20)] = VollathTest(im(:, :, 3));
62   %[maxF, imaxF] = max(Focus(1:i, 4))
63   plot(Focus(1:i, 1), Focus(1:i, 6), 'r', Focus(1:i, 1), Focus(1:i, 7), 'g', Focus
        (1:i, 1), Focus(1:i, 8), 'b');
64   end
65   %After obtain all FM VALUES, find the MaxFM VALUE and its position
66   %within the FOCUS structure
67   [maxF, imaxF] = max(Focus(1:i, 4));
68   % GOING BACK TO THE BEST POSITION
69   d2=0;
70   for n= imaxF+1:Cycles
71       abs(Focus(n, 6)-Focus(n-1, 7));
72       d2= d2 + abs(Focus(n, 6)-Focus(n-1, 7));
73   end
74   d1= abs(Focus(imaxF, 6)-Focus(imaxF, 7))-abs(Focus(imaxF, 2));
75
76   GoingBack=(-(imaxF-Cycles)*delta)+d2+d1
77   %TeleMove_10May(leo, GoingBack);
78   end

```

Appendix B

Features Camera

The camera model and its features is found here, in this project is used the the Point Gray camera with BFLY-U3-23S6C-C part number.

FLIR **BLACKFLY**[®] USB3 VISION

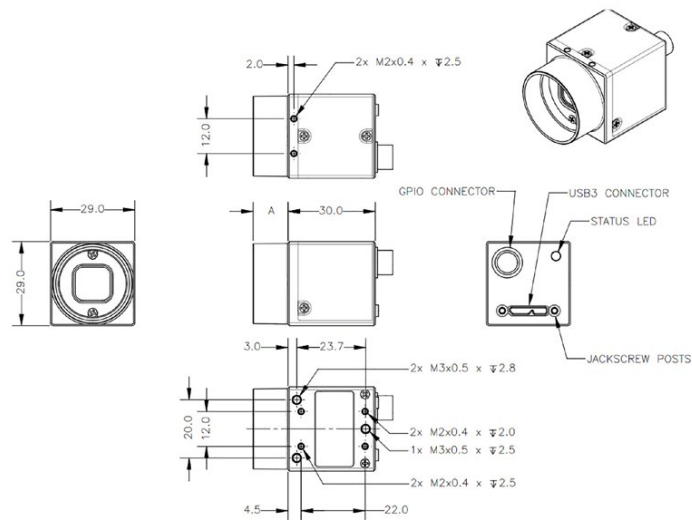


UNBEATABLE VALUE + ULTRA-COMPACT

The Blackfly USB3 Vision camera line combines the throughput of USB 3.0, highly sensitive sensors, industry-leading affordability, and a host of unique features for uncompromising value. The ultra-compact Blackfly camera weighs just 36 grams, measures 29 x 29 x 30mm, uses 3 watts of power, and comes with a 3-year warranty.

KEY FEATURES

- Perfect analog camera replacement, only 29 x 29 x 30mm
- On-camera image processing: color interpolation, gamma, and LUT
- 16 MByte frame buffer; LED status indicator



www.flir.com/mv



The World's **Sixth Sense**[®]

FIGURE B.1: Texto para la leyenda

Specifications

Model	Version	MP	Imaging Sensor
BFLY-U3-03S2C-CS BFLY-U3-03S2M-CS	Color Mono	0.3 MP	<ul style="list-style-type: none"> ■ Sony ICX424, 1/3", 7.4 μm ■ Global shutter ■ 84 FPS at 648 x 488
BFLY-U3-05S2C-CS BFLY-U3-05S2M-CS	Color Mono	0.5 MP	<ul style="list-style-type: none"> ■ Sony ICX693, 1/3", 6.0 μm ■ Global shutter ■ 50 FPS at 808 x 608
BFLY-U3-13S2C-CS BFLY-U3-13S2M-CS	Color Mono	1.3 MP	<ul style="list-style-type: none"> ■ Sony ICX445, 1/3", 3.75 μm ■ Global shutter ■ 30 FPS at 1288 x 964
BFLY-U3-20S4C-CS BFLY-U3-20S4M-CS	Color Mono	2.0 MP	<ul style="list-style-type: none"> ■ Sony ICX274, 1/1.8", 4.4 μm ■ Global shutter ■ 15 FPS at 1624 x 1224
BFLY-U3-23S6C-C BFLY-U3-23S6M-C	Color Mono	2.3 MP	<ul style="list-style-type: none"> ■ Sony IMX249, 1/1.2", 5.86 μm ■ Global shutter ■ 41 FPS at 1920 x 1200
BFLY-U3-50H5C-C BFLY-U3-50H5M-C	Color Mono	5.0 MP	<ul style="list-style-type: none"> ■ Sharp RJ32S3AA00DT, 2/3", 3.45 μm ■ Global shutter ■ 7.5 FPS at 2448 x 2048

FIGURE B.2: Blacky USB3 Vision Camera Specifications

Imaging Performance (EMVA 1288)	See the Imaging Performance Specification , which includes quantum efficiency, saturation capacity (full well depth), read noise, dynamic range and signal to noise ratio.
A/D Converter	12-bit; 10- and 12-bit (BFLY-U3-23S6)
Video Data Output	8, 12, 16 and 24-bit digital data
Image Data Formats	Mono8, Mono12, Mono16, Raw8, Raw12, Raw16 (all models); RGB, YUV411, YUV422, YUV 444 (color models)
Partial Image Modes	Pixel binning, decimation, and region of interest (ROI) modes
Image Processing	Gamma, lookup table, hue, saturation, and sharpness
Shutter	Global shutter; Automatic/manual/one-push See product webpage for specific model's range
Gain	Automatic/manual/one-push See product webpage for specific model's range
Gamma	0.50 to 3.99, programmable lookup table
White Balance	Automatic/manual/one-push
Color Processing	On-camera in YUV or RGB format, or on-PC in Raw format
Digital Interface	USB 3.0 interface with screw locks for camera control, data, and power
Transfer Rates	5 Gbit/s
GPIO	6-pin Hirose HR10A-7R-6PB GPIO connector for trigger, strobe, and power
External Trigger Modes	Standard, bulb, low smear, overlapped, multi shot
Image Buffer	16 MB frame buffer
Memory Channels	2 user configuration sets for custom camera settings
Flash Memory	1 MB non-volatile memory
Dimensions	29 mm x 29 mm x 30 mm excluding lens holder, without optics (metal case)
Mass	36 g (without optics)
Power Consumption	5-24 V via GPIO or 5 V via USB 3.0 interface, maximum <3 W
Machine Vision Standard	USB3 Vision
Camera Control	Via FlyCapture SDK or USB Vision third party software
Camera Updates	In-field firmware updates
Lens Mount	CS-mount; C-mount (BFLY-U3-23S6 and BFLY-U3-50H5)
Temperature	Operating: 0° to 45°C; Storage: -30° to 60°C
Humidity	Operating: 20 to 80% (no condensation); Storage: 20 to 95% (no condensation)
Compliance	CE, FCC, KCC, RoHS
Operating System	Windows, Linux (32- and 64-bit)
Warranty	3 years

FIGURE B.3: Blacky USB3 Vision camera General features

Appendix C

Hardware

A complete circuit of the hardware part will be presented in this part, to show all the configuration.

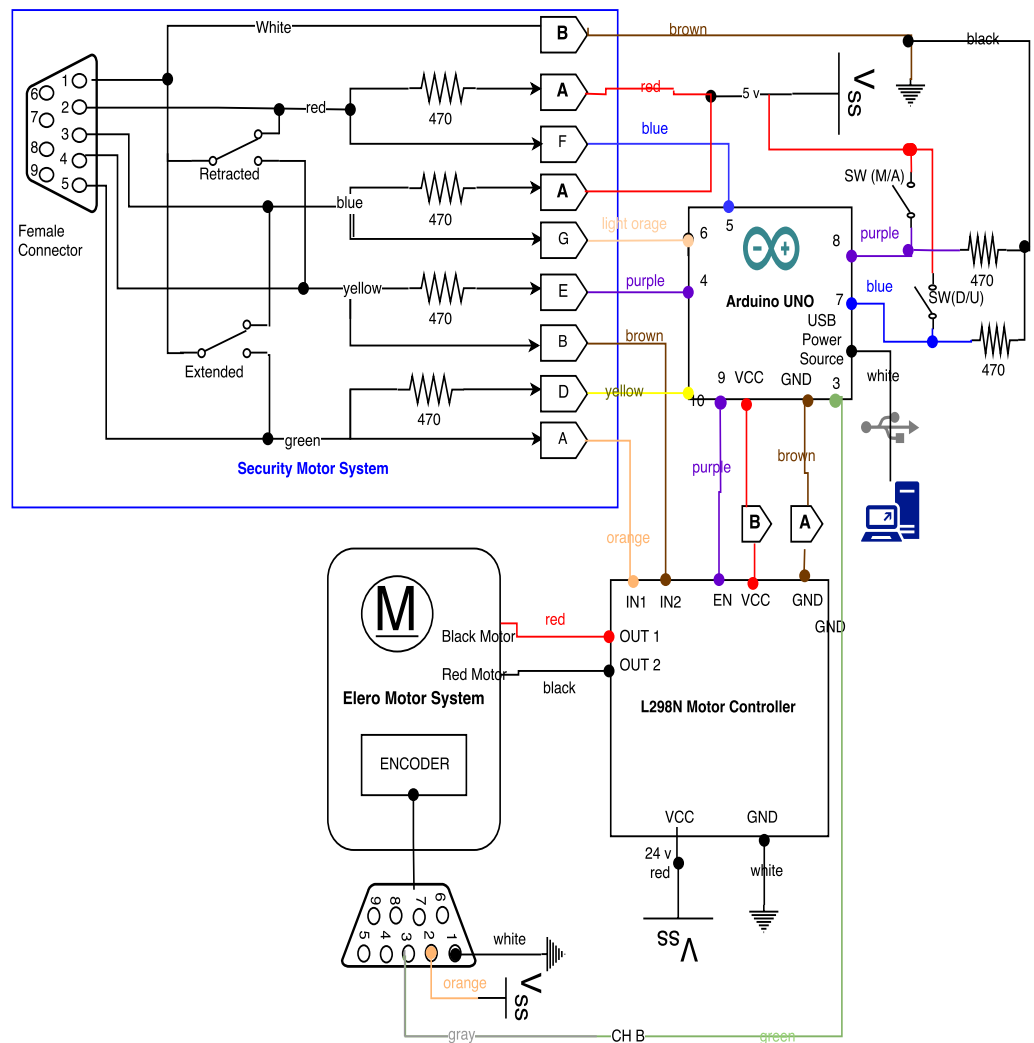


FIGURE C.1: External circuit part of the hardware movement control