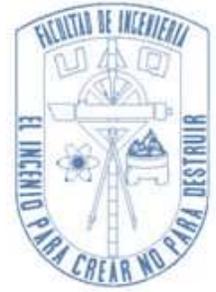




**UNIVERSIDAD AUTÓNOMA DE QUERÉTARO**  
**FACULTAD DE INGENIERÍA**  
**MAESTRÍA EN CIENCIAS MECATRÓNICA**



Desarrollo de prototipo para detección de obstáculos en personas  
invidentes por medio de visión artificial

**TESIS**

Que como parte de los requisitos para obtener el título de  
Maestro en Ciencias, especialidad Mecatrónica

Presenta:

**Ing. Salvador Martínez Cruz**

Dirigido por:

**Dr. Luis Alberto Morales Hernández**

San Juan del Río, Querétaro

Enero 2019



**Universidad Autónoma de Querétaro**  
**Facultad de Ingeniería**  
**Maestría en Ciencias Mecatrónica**

**Desarrollo de prototipo para detección de obstáculos en  
personas invidentes por medio de visión artificial**

Opción de titulación  
**Tesis**

Que como parte de los requisitos para obtener el Grado de  
Maestro en Ciencias Mecatrónica

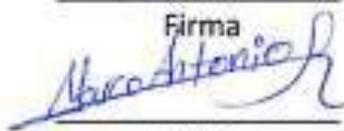
**Presenta:**  
Ing. Salvador Martínez Cruz

**Dirigido por:**  
Dr. Luis Alberto Morales Hernández

Dr. Luis Alberto Morales Hernández  
Presidente

  
\_\_\_\_\_  
Firma

M. en C. Marco Antonio Garduño Ramón  
Secretario

  
\_\_\_\_\_  
Firma

Dr. Gerardo Israel Pérez Soto  
Vocal

  
\_\_\_\_\_  
Firma

Dr. Martín Valtierra Rodríguez  
Suplente

  
\_\_\_\_\_  
Firma

Dr. Juan Primo Benítez Rangel  
Suplente

  
\_\_\_\_\_  
Firma  
\_\_\_\_\_  
Dr. Manuel Toledano Ayala  
Director de la Facultad Ingeniería  
\_\_\_\_\_  
Dra. Ma. Guadalupe Flavia Loarca Piña  
Directora de Investigación y Posgrado

Centro Universitario  
Querétaro, Qro.  
Fecha (será el mes y año de aprobación del Consejo Universitario)

## RESUMEN

En este documento se describe el desarrollo de un sistema de visión artificial portátil que tiene como principal objetivo ser implementado en una persona con debilidad visual o ciega para darle una noción de lo que se encuentra en su entorno. Se lleva a cabo en 4 etapas principales que son; la extracción de información, es donde se obtiene la información del entorno o del espacio de trabajo mediante la captura de imágenes desde dos cámaras combinado con la distancia hacia el objeto más cercano que se obtiene con un sensor ultrasónico. El procesamiento de las imágenes, se aplican varias herramientas de tratamiento de imágenes como la escala de grises, la corrección y la rectificación para tener información certera del espacio de trabajo. Mapa de profundidad, con las imágenes obtenidas se aplican algoritmos de estereovisión para saber qué objeto se encuentra más cerca que otro con respecto al usuario. La traducción de la información, se desarrolla una metodología matemática para traducir la información obtenida en señales de frecuencia que posteriormente se implementan en dispositivos de vibración mecánica para que, de esta forma, el usuario tenga noción del entorno que lo rodea.

(Palabras clave; Visión artificial, dispositivo, debilidad visual, procesamiento de imágenes).

## **SUMMARY**

This document describes the development of a portable machine vision system whose main objective is to be implemented in a person with visual weakness or blindness to give a sense of what is in their environment. It takes place in 4 main stages; the extraction of information, is where the information of the environment or workspace is obtained by capturing images from two cameras combined with the distance to the nearest object obtained with an ultrasonic sensor. The processing of images, several image processing tools such as grayscale, correction and rectification are applied to have accurate information of the workspace. Depth map, with the obtained images, stereovision algorithms are applied to know which object is closer than another with respect to the user. The translation of the information, a mathematical methodology is developed to translate the information obtained into frequency signals that are subsequently implemented in mechanical vibration devices, in this way, the user has a sense of the surrounding environment.

(Keywords: artificial vision, device, visual weakness, image processing).

## DEDICATORIA

Para mi madre y hermanos que me han apoyado en todo momento a lo largo de mi vida.

## **AGRADECIMIENTOS**

Agradezco a Dios por darme la vida, la salud y los recursos económicos para poder concluir esta etapa de mi vida.

A mi madre por darme la vida, el amor, la corrección, la educación, los buenos valores y sobre todo por ser la persona más trabajadora, correcta, humilde y perseverante que he conocido. Por apoyarme siempre en el ámbito emocional y económico que ha sido indispensable para llegar a esta etapa de mi vida.

A mis hermanos y hermanas por siempre estar dispuestos a ayudarme en lo que he necesitado, por confiar en mi y por el gran ejemplo de personas que son.

Le doy un agradecimiento especial al Dr. Luis Alberto Morales Hernández que más allá de ser mi asesor de tesis es un buen amigo y una persona de la que he aprendido mucho. Gracias por su apoyo para revisar mis avances de tesis por todos los consejos que me ha dado sobre el desarrollo de nuestro proyecto, por nunca dejarme solo cuando algo no está funcionando o cuando no se obtiene lo que se espera. De usted he aprendido que la paciencia rinde muchos frutos, que si algo no funciona se arregla mas no se hace a un lado. Le agradezco por el apoyo laboral, moral y económico que siempre me ha brindado y espero seguir trabajando con usted en la siguiente etapa.

Agradezco al Dr. Gerardo Israel Pérez Soto por estar conmigo desde la licenciatura, por estar atento al desarrollo de nuestro proyecto, porque a pesar de tener muchos asesorados se da el tiempo para atendernos a todos y está siempre pendiente. Gracias por todo lo que me ha enseñado no sólo lo académico sino por enseñarme a mejorar mi comportamiento, a controlarme cuando algo me molesta, a cuidar mucho lo que digo y a mejorar mi manera de pedir un favor. Son muchas las cosas que he aprendido de usted muchas gracias.

Gracias al Dr. Marco Antonio Garduño Ramón por apoyarme en el desarrollo de este proyecto, por la paciencia que me ha tenido ya que a veces nada de lo que me pedía que hiciera me funcionaba, pero en vez de molestarse me lo explicaba de

nuevo. Gracias por apoyarme en el desarrollo del software que es la parte medular de esta tesis.

Agradezco al Dr. Martín Valtierra Rodríguez y al Dr. Juan primo Benítez Rangel por el apoyo en la revisión del documento final de esta tesis, así como por darme sus observaciones y consejos en las presentaciones previas.

Al Dr. Roque A. Osornio Ríos por brindarme la oportunidad y confianza de realizar mis estudios de posgrado en la UAQ, poniendo sus conocimientos y tiempo a mi disposición a lo largo de mi estancia en la universidad.

Al Consejo Nacional de Ciencia y Tecnología CONACyT por el apoyo económico proporcionado a través de la beca No. 783323, para la realización de mis estudios de posgrado en la Universidad Autónoma de Querétaro.

A la Universidad Autónoma de Querétaro, por brindarme la oportunidad de realizar mis estudios y proporcionarme las herramientas y el apoyo necesario para el desarrollo de este trabajo.

# ÍNDICE GENERAL

<b>1. INTRODUCCIÓN...</b>	<b>1</b>
1.1. ANTECEDENTES...	2
1.1.1. TRABAJOS PARA AYUDAR A PERSONAS DÉBILES VISUALES...	2
1.1.2. TRABAJOS CON RASPBERRY PI Y LIBRERÍAS OPENCV...	3
1.1.3. TRABAJOS DONDE SE IMPLEMENTA VISIÓN ESTEREOSCÓPICA...	3
1.2. JUSTIFICACIÓN...	4
1.3. DESCRIPCIÓN DEL PROBLEMA...	4
1.4. HIPÓTESIS...	4
1.5. OBJETIVOS...	4
1.5.1. OBJETIVO PRINCIPAL...	5
1.5.2. OBJETIVOS PARTICULARES...	5
<b>2. FUNDAMENTOS TEÓRICOS...</b>	<b>6</b>
2.1. CONCEPTOS DE PROGRAMACIÓN APLICADOS A SISTEMAS DE VISIÓN...	6
2.1.1. CONVERSIÓN A ESCALA DE GRISES...	6
2.1.2. OPERACIONES MORFOLÓGICAS...	6
2.1.2.1. DILATACIÓN...	7
2.1.2.2. EROSIÓN...	9
2.2. VISIÓN ESTEREOSCÓPICA...	10
2.3. AMPLIFICACIÓN DE POTENCIA...	12
2.3.1. MODULACIÓN POR ANCHO DE PULSO...	13
2.3.2. AMPLIFICACIÓN DE SEÑALES CON TRANSISTOR...	14
2.4. DESCRIPCIÓN DE HARDWARE...	16
2.4.1. SENSOR...	16
2.4.2. SENSOR DE PROXIMIDAD...	16
2.4.3. SENSOR DE PROXIMIDAD DE ULTRASONIDO...	16
2.4.4. TRANSDUCTOR...	17

2.4.5. ACTUADORES DE VIBRACIÓN MECÁNICA...	...	...	...	...	17
<b>3. DESARROLLO DE UN SISTEMA DE VISIÓN ESTEREOSCÓPICA...</b>					
...	...	...	...	...	<b>19</b>
3.1. DISEÑO DE SOPORTE DE CÁMARAS...	...	...	...	...	20
3.2. AJUSTE DE DISTANCIA ENTRE CÁMARAS...	...	...	...	...	22
3.3. ADQUISICIÓN DE IMÁGENES...	...	...	...	...	23
3.4. CALIBRACIÓN DE CÁMARAS...	...	...	...	...	24
<b>4. CONVERSIÓN DE INFORMACIÓN TRIDIMENSIONAL A DIAGRAMA DE FRECUENCIA...</b>	...	...	...	...	<b>27</b>
<b>5. DESARROLLO DE UN SISTEMA PARA DETECCIÓN DE OBSTÁCULOS APLICANDO VISIÓN ARTIFICIAL...</b>	...	...	...	...	<b>30</b>
5.1. IMPLEMENTACIÓN DE ACTUADORES DE VIBRACIÓN...	...	...	...	...	33
5.2. PRUEBAS DEL DISPOSITIVO Y DISCUSIÓN DE RESULTADOS...	...	...	...	...	34
<b>6. CONCLUSIONES...</b>	...	...	...	...	<b>38</b>
<b>7. REFERENCIAS...</b>	...	...	...	...	<b>40</b>
<b>APÉNDICE 1. CÓDIGO DE CALIBRACIÓN DE CÁMARAS...</b>	...	...	...	...	<b>42</b>
<b>APÉNDICE 2. CÓDIGO DE MAPA DE PROFUNDIDAD...</b>	...	...	...	...	<b>44</b>

## ÍNDICE DE FIGURAS

Figura 2.1. Imagen original donde se aplican operadores morfológicos... ..	7
Figura 2.2. Imagen binarizada... ..	8
Figura 2.3. Imagen dilatada... ..	9
Figura 2.4. Imagen erosionada... ..	10
Figura 2.5. Diagrama representativo de la visión estereoscópica... ..	11
Figura 2.6. Diagrama representativo del proceso informático para aplicar visión estereoscópica... ..	12
Figura 2.7. Diagrama representativo de la modulación por ancho de pulso...	13
Figura 2.8. Representación gráfica de la función de amplificación... ..	15
Figura 2.9. a) Diagrama del transistor PNP, b) Diagrama del transistor NPN...	15
Figura 2.10. Motor de vibración mecánica GlobalSources (2016) ... ..	17
Figura 3.1. Diagrama general de la metodología seguida para el desarrollo del presente proyecto de tesis... ..	19
Figura 3.2. Diagrama del procesamiento de imágenes para aplicar visión estereoscópica... ..	20
Figura 3.3 Diseño de base para sostener las cámaras... ..	21
Figura 3.5. Diagrama representativo de la ecuación de disparidad... ..	23
Figura 3.7. Imágenes capturadas para la calibración de las cámaras... ..	25
Figura 3.8. Figuras corregidas con la calibración... ..	26
Figura 4.1. Gráficas de las funciones traducidas del espacio al diagrama de frecuencia... ..	28
Figura 5.1. Base para sostener las cámaras a la distancia adecuada... ..	30
Figura 5.2. Montaje de cámaras a la distancia adecuada... ..	31
Figura 5.3. Captura de imágenes para obtención de profundidad... ..	32
Figura 5.4. Puntos de correspondencia en ambas imágenes... ..	32
Figura 5.5. Imágenes rectificadas con puntos de coincidencia corregidos...	33
Figura 5.6. Colocación de actuadores de vibración... ..	34

Figura 5.7. Comprobación de linealidad en la frecuencia de vibración... ..	34
Figura 5.8. Gráficas de las funciones traducidas del espacio al diagrama de frecuencia... ..	36

## ÍNDICE DE TABLAS

Tabla 2.1. Especificaciones de los moto-vibradores empleados...	...	...	18
Tabla 4.1. Fórmulas de traducción de información...	...	...	27
Tabla 4.2 representación de las variables en las fórmulas desarrolladas...	...	...	28
Tabla 5.1. Resultados de la calibración de las cámaras...	...	...	35
Tabla 5.2. Valores de entrada en la comprobación de las fórmulas desarrolladas...	...	...	36
Tabla 6.1. Fórmulas de traducción de información...	...	...	38
Tabla 6.2 representación de las variables en las fórmulas desarrolladas...	...	...	38

## ÍNDICE DE ECUACIONES

Ecuación (2.1) ...	...	...	...	...	...	...	...	...	6
Ecuación (2.2) ...	...	...	...	...	...	...	...	...	13
Ecuación (2.3) ...	...	...	...	...	...	...	...	...	14
Ecuación (3.1) ...	...	...	...	...	...	...	...	...	22

# 1. INTRODUCCIÓN

Durante muchos años, uno de los objetivos principales del ser humano es proteger su especie, adaptarla a nuevos cambios del mundo moderno y hacer que su estadía en este mundo sea cada vez más placentera y duradera. Por medio de la tecnología el hombre ha podido mejorar su calidad de vida, en especial, a personas que poseen alguna discapacidad para mejorar su desempeño en el desarrollo de ciertas tareas. Por ejemplo, una silla de ruedas sirve para personas que no puedan desplazarse por sí solas, un bastón o un perro lazarillo sirven para que personas invidentes puedan transitar por la calle evitando obstáculos o cualquier otro tipo de situación peatonal. La automatización y la electrónica han jugado un papel muy importante en el desarrollo de varios dispositivos de ayuda para gente que posee alguna falencia, ya que actualmente se hacen mejoras en aparatos para terapias en alguna extremidad en particular, en el reemplazo parcial o total de algún miembro o parte del cuerpo humano, también en el desarrollo de máquinas y dispositivos para mejorar la calidad de vida de dichos pacientes.

Al ver un sin fin de posibilidades que poseen actualmente las personas con alguna discapacidad, se puede destacar que los invidentes mexicanos no tienen una tecnología en particular que pueda facilitar su desplazamiento por las calles, debido a que siempre se les puede observar con sus aparatos tradicionales como el bastón o el perro guía. Por esta razón es necesario aplicar alguna tecnología para la ayuda en la deambulación de personas con ceguera o debilidad visual, que se puedan dar cuenta que hay tecnología disponible y cosas por hacer.

El instrumento que se obtuvo en esta tesis, es un dispositivo para la detección de obstáculos mediante visión artificial, si bien, existen diversos dispositivos parecidos en Europa, estos son en su mayoría de alto costo por lo que un invidente común no puede acceder a este tipo de tecnologías. Este dispositivo no reemplaza totalmente instrumentos de ayuda común en la deambulación de personas con discapacidad visual, pero dará apoyo a este tipo de implementos y en ocasiones sustituirlos temporalmente según la actividad que se esté desarrollando por parte del usuario.

## 1.1.ANTECEDENTES

En esta sección se presentan los trabajos más influyentes desarrollados en los últimos años que están directamente relacionados con el presente proyecto de Tesis.

### 1.1.1. TRABAJOS PARA AYUDAR A PERSONAS DÉBILES VISUALES

Zereen et al (2016) desarrollaron una técnica para detectar el movimiento de un objeto junto con su dirección. La profundidad hacia el objeto es tomada mediante un kinect, después se hace el procesamiento de la información para finalmente transmitirla al usuario. Rashid et al (2016) desarrollaron un sistema de asistencia bilingüe portable para personas con discapacidad visual. Es un dispositivo que les permite identificar los colores de un objeto y se transmite por audio en dos lenguajes; bengalí o inglés. La información se transmite mediante audífonos ya sea alámbricos o por bluetooth. El procesamiento lo lleva a cabo usando un microcontrolador Arduino Nano, un módulo bluetooth HC-05, y un sensor de color TCS 3200. Por su parte Everding et al (2016) presentaron un trabajo en el cual desarrollan un dispositivo ligero y portable para ayudar a personas visualmente discapacitadas en sus actividades cotidianas. Utilizaron sensores de visión dinámica inspirados en retina para recopilar la información, aplicaron algoritmos basados en eventos para extraer y operar la información de profundidad en tiempo real para posteriormente traducirla al dominio acústico y de esta forma dar al usuario la noción del entorno que lo rodea. También Hejun et al (2015) presentaron un trabajo en el cual se desarrolla un dispositivo que asemeja el funcionamiento de un bastón para las personas invidentes. Consiste en la conexión de varios sensores ultrasónicos que detectan la proximidad hacia los objetos cercanos en diferentes direcciones y esta señal se transmite a los usuarios mediante señales de audio y de vibración como alertas ON-OFF.

Ghaderi et al (2015) propusieron un dispositivo que ayude a las personas invidentes a desplazarse con facilidad evitando los obstáculos que se les presentan. Para esto utilizaron dos sensores de visión dinámica y técnicas de procesamiento de información basadas en eventos para extraer información de profundidad y convertirlo en sonido tridimensional de salida utilizando un sistema individualizando

la función de transferencia relacionada con la cabeza. Otro trabajo parecido es el que presentan Mohamed et al (2015) en el que se pretende ayudar a las personas ciego-sordo-mudas en el cual usaron sensores ultrasónicos para detectar obstáculos en frente, detrás y debajo de las personas y que a su vez se les informa mediante vibraciones mecánicas que sirven como advertencia. Ramírez-Garibay et al (2014) desarrollaron un teclado portátil con braille y un altavoz para la comunicación entre dos personas ciegas. Por otra parte, Raghuvanshi et al (2014) presentaron un dispositivo que sirve como auxiliar a personas invidentes para poder detectar los objetos en su entorno ya que cuando está a punto de tener contacto con él produce un sonido que sirve como advertencia, esto se logra mediante el uso de sensores ultrasónicos.

#### 1.1.2. TRABAJOS CON RASPBERRY PI Y LIBRERÍAS OPENCV

Yin et al (2016) desarrollaron un software de reconstrucción facial en tres dimensiones, utilizaron cámaras zhongxing-micro ZC301P con un sistema de visión estereoscópica binocular para grabar imágenes. Estas imágenes son extraídas utilizando las funciones de las librerías de OpenCV y posteriormente se hace la reconstrucción. Sung et al (2016) presentaron un programa en que se hace un flujo de optimización para aplicaciones de reconocimiento de gestos utilizando las librerías de OpenCV. Hossain et al (2015) presentaron un sistema de vigilancia de bajo costo que pretende enviar imágenes en tiempo real con el objetivo de mantener la seguridad en las instalaciones donde se utilice. Para ello usaron una tarjeta Raspberry pi 2 y en el software implementaron las librerías de OpenCV. Patchava et al (2015) propusieron un sistema de automatización residencial con Raspberry Pi. Se hace mediante la integración de cámaras y sensores de movimiento en una aplicación web.

#### 1.1.3. TRABAJOS DONDE SE IMPLEMENTA VISIÓN ESTEREOSCÓPICA

Gehrig et al (2016) desarrollaron un sistema de visión estereoscópica robusto, que pueda funcionar en ambientes externos, con lluvia o nieve. Implementan memoria de eventos pasados para tener información correcta. Por otra parte, Häne et al (2016) desarrollaron un sistema montable para conducción autónoma de automóviles basado en visión estereoscópica.

## 1.2.JUSTIFICACIÓN

Actualmente no existen sistemas tecnológicos de bajo costo para ayudar a personas con discapacidad que les permita disfrutar de igualdad de oportunidades, así como una participación plena y efectiva en la sociedad en aspectos de la vida. En este trabajo se pretende desarrollar un dispositivo que ayude a desempeñar las actividades cotidianas a personas que padecen de una debilidad visual y, además, que reduzca el riesgo de sufrir un accidente o perder la vida.

La mayoría de las herramientas de apoyo a personas débiles visuales trabajan como una advertencia ON-OFF, es decir, advierten a la persona si hay o no hay algo que se interponga en su camino, en cambio con este trabajo la persona estará informada si hay algún obstáculo en su camino y también tendrá noción de la distancia a la que éste se encuentra y de la forma que tiene.

## 1.3.DESCRIPCIÓN DEL PROBLEMA

De acuerdo con La Organización de Naciones Unidas (ONU), las personas con discapacidad en el mundo son más de mil millones o el 15 por ciento de la población del mundo. En México hay más de 6 millones de personas discapacitadas de acuerdo con el Instituto Nacional de Estadística INEGI (2016).

Se han desarrollado recientemente aplicaciones en dispositivos móviles y dispositivos montables que ayudan en algunas de las tareas cotidianas para los débiles visuales, sin embargo, no se ha trabajado profundamente en el desarrollo de tecnología para darle a la persona una noción acertada de lo que tiene a su alrededor.

## 1.4.HIPÓTESIS

Es posible conocer la distancia, forma y tamaño de los objetos que se encuentran en un espacio de trabajo determinado mediante visión estereoscópica y transmitir la información a una persona mediante una combinación de señales de vibración mecánica.

## 1.5.OBJETIVOS

En esta sección se describen consistentemente los objetivos del presente proyecto de tesis.

#### 1.5.1. OBJETIVO PRINCIPAL

Desarrollar un prototipo portátil que obtenga información tridimensional mediante captura de imágenes en dos dimensiones con algoritmos de visión estereoscópica y traduzca esta información en señales digitales de frecuencia para convertirlas en señales de vibración mecánica y darle una noción del entorno a una persona con debilidad visual o ciega.

#### 1.5.2. OBJETIVOS PARTICULARES

Implementar herramientas de visión artificial como escala de grises, filtros de corrección y calibración para obtener un mapa de profundidad del espacio de trabajo.

Diseñar y construir un sistema portátil que incluya dos cámaras y un sensor ultrasónico de forma que lo pueda llevar montado una persona en la cabeza.

Proponer un modelo matemático que traduzca la información de profundidad a un diagrama de frecuencia para que la persona pueda estar consciente de lo que pasa a su alrededor.

## 2. FUNDAMENTOS TEÓRICOS

En esta sección se describen brevemente las herramientas que se utilizarán en este trabajo.

### 2.1. CONCEPTOS DE PROGRAMACIÓN APLICADOS A SISTEMAS DE VISIÓN

En esta sección se presentan los conceptos fundamentales para hacer procesamiento digital de imágenes.

#### 2.1.1. CONVERSIÓN A ESCALA DE GRISES

La manera más simple, de convertir una imagen en color a blanco y negro, es a lo que todo programa de retoque trae de "convertir a escala de grises". La forma en que el programa convierte a grises, aunque parezca un misterio, no lo es tanto: en realidad hace un mezclado de los tres canales (RGB) para obtener uno solo en gris tomando los siguientes porcentajes: Rojo 30%, Verde 59%, Azul 11%, que es según los expertos lo más parecido a como nuestros ojos captan la intensidad de luz dependiendo del color de ésta. La Ecuación 2.1 nos muestra la expresión matemática de ese fenómeno y los factores de ponderación de cada componente de color nos indican la sensibilidad del ojo humano a las frecuencias del espectro cercanas al rojo, verde y azul, como sigue:

$$y = R * 0.3 + G * 0.59 + B * 0.11 \quad (2.1)$$

Por tanto, para realizar esta conversión basta con aplicar esta ecuación a cada pixel de la imagen RGB. Aparecería una nueva matriz de un byte por pixel que daría la información de luminancia. Para convertir a escala de grises una imagen de paleta, basta con convertir cada color de la paleta a su gris correspondiente según la ecuación anterior, creando una nueva paleta de grises.

#### 2.1.2. OPERACIONES MORFOLÓGICAS

A continuación, se explican las operaciones morfológicas mayormente utilizadas en el procesamiento digital de imágenes.

### 2.1.2.1. DILATACIÓN

Cuando se termina de delimitar objetos en una imagen usando los operadores de búsqueda de bordes aparece el problema de definir dichos objetos dentro del sistema.

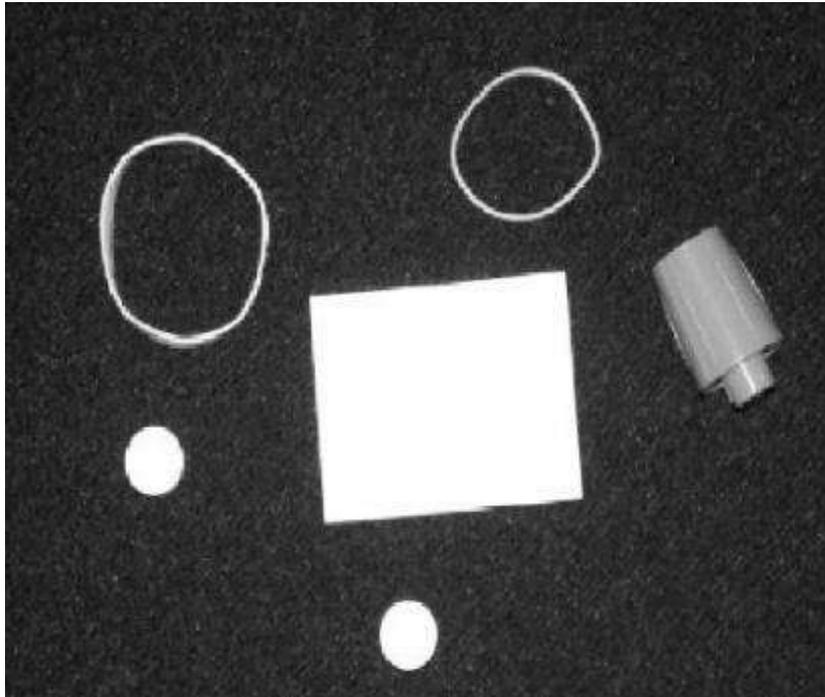


Figura 2.1. Imagen original donde se aplican operadores morfológicos.

Supongamos que definimos una característica que definió cual pixel está adentro y cual está afuera de cada objeto. Creamos así imágenes binarias (un bit para la cuantificación) blanco 0 negro 1.

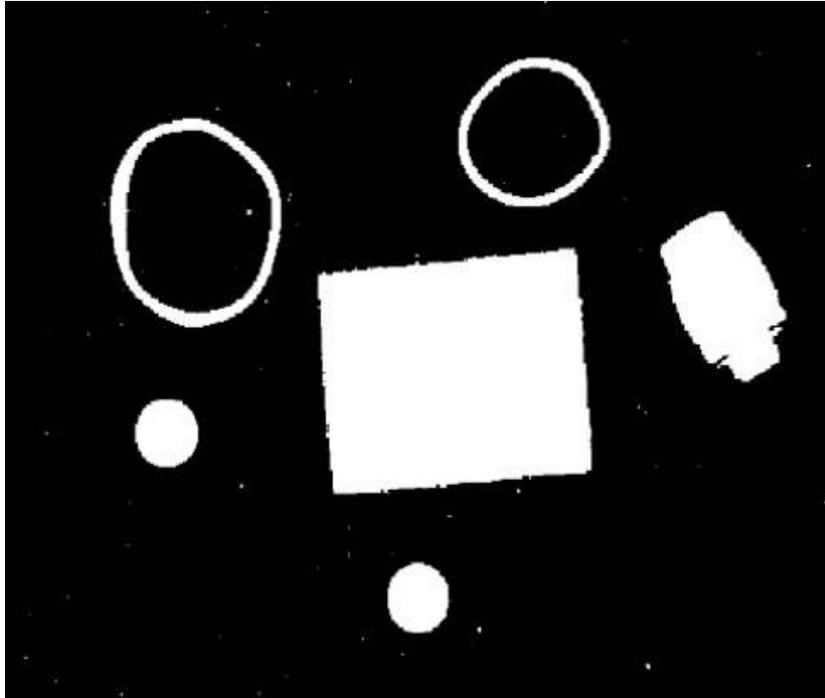


Figura 2.2. Imagen binarizada.

Obtenida la imagen binaria se ve que quedan pixeles o pequeños grupos de ellos que sabemos que deberían pertenecer al objeto pero que por diferentes motivos no quedaron identificados. Otra posibilidad es que aparezcan pixeles que sabemos que no deberían haber aparecido.

Para incorporar estos pixeles al objeto conviene hacer una operación que se corresponde al siguiente concepto; Si alguno de los pixeles vecinos al pixel en estudio pertenece al objeto entonces el pixel de estudio también pertenece al objeto. A esta operación lógica se la llama operación de dilatación.

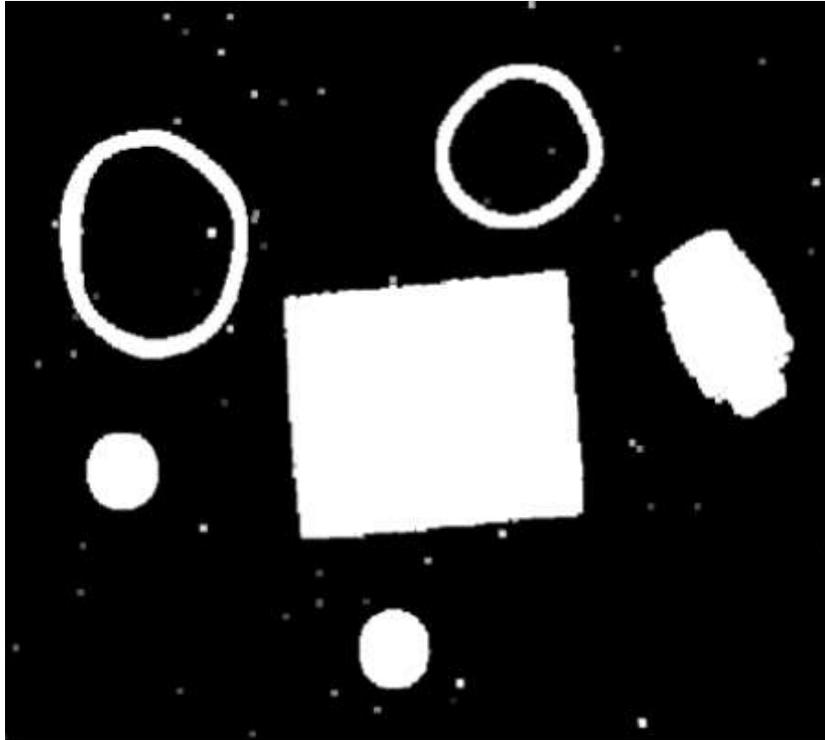


Figura 2.3. Imagen dilatada.

#### 2.1.2.2. EROSIÓN

Para quitar pixeles que no pertenecen al objeto conviene hacer otra operación; Si todos los pixeles vecinos al pixel de estudio pertenecen al objeto, entonces el pixel de estudio también pertenece al objeto. (Si alguno de los pixeles vecinos al pixel de estudio no pertenece al objeto entonces ese pixel de estudio tampoco pertenece al nuevo objeto). Esta operación se llama erosión.

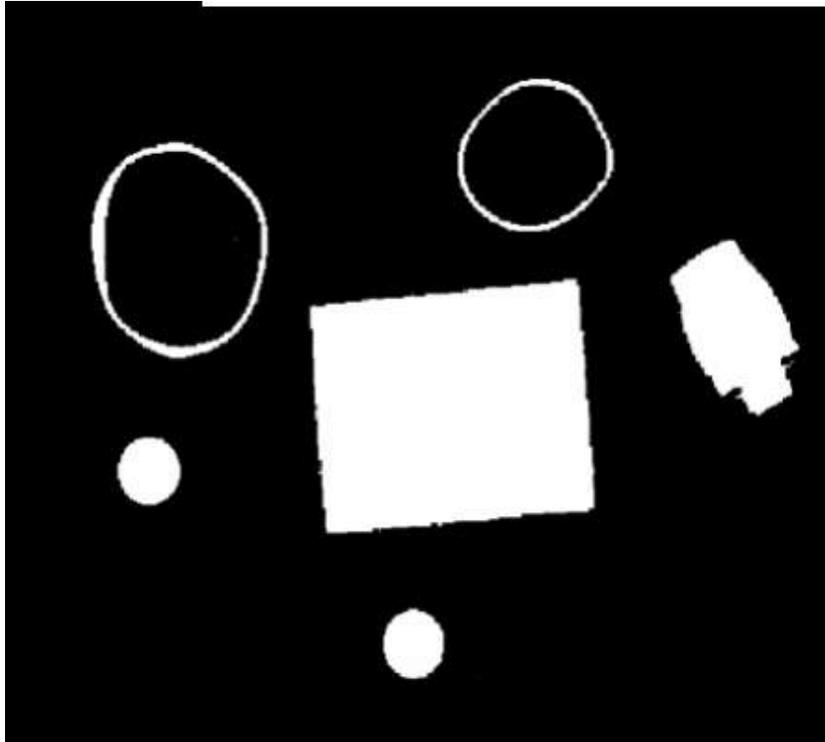


Figura 2.4. Imagen erosionada.

## 2.2.VISIÓN ESTEREOSCÓPICA

La visión estereoscópica es una facultad físico - psicológica que posee el ser humano, que le permite ver en tercera dimensión aquellos objetos que contempla mediante su visión binocular. Sobre cada una de las retinas de ambos ojos, se forma una imagen perspectiva del mismo objeto, las cuales difieren entre sí, debido a la distinta posición de los puntos de vista, produciendo el efecto de relieve. La distancia entre estos dos puntos de vista, o sea, la separación entre los órganos receptores del ser humano, tiene un valor promedio de 65 mm y se la llama "distancia interpupilar". Cada uno de los ejes ópticos de los cristalinos rotan en el interior de su órbita, hasta que sus direcciones intersecan el punto P de la Figura 2.5 en cuestión, proceso éste que llamaremos "convergencia", tal que las imágenes P' y P" se forman en una pequeña fosa existente en las retinas (mancha amarilla).

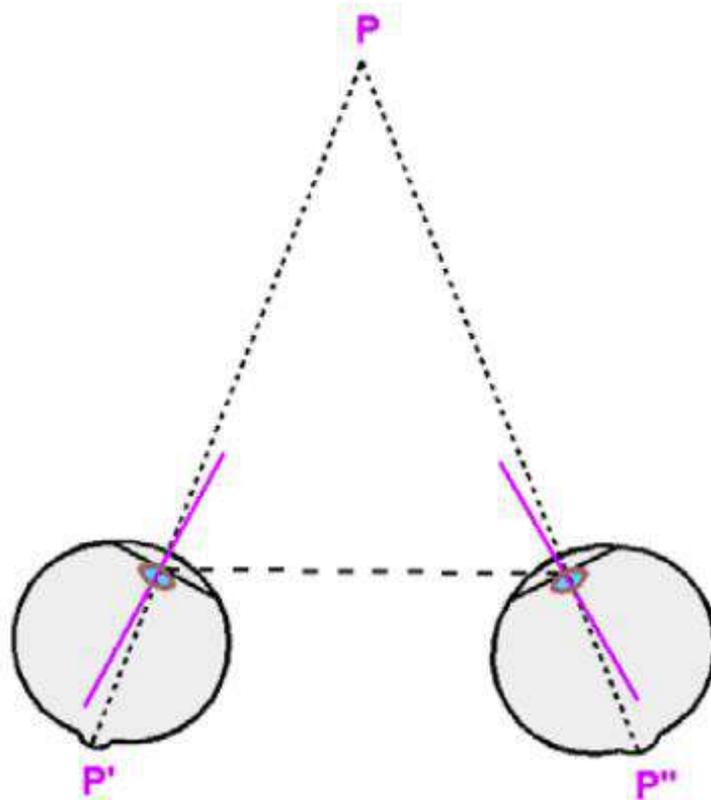


Figura 2.5. Diagrama representativo de la visión estereoscópica.

La visión estereoscópica implica la extracción de información de profundidad desde dos vistas diferentes de una escena obtenida por dos cámaras diferentes. Para lograr obtener la información en tercera dimensión se sigue la metodología mostrada en el diagrama de la Figura 2.6.

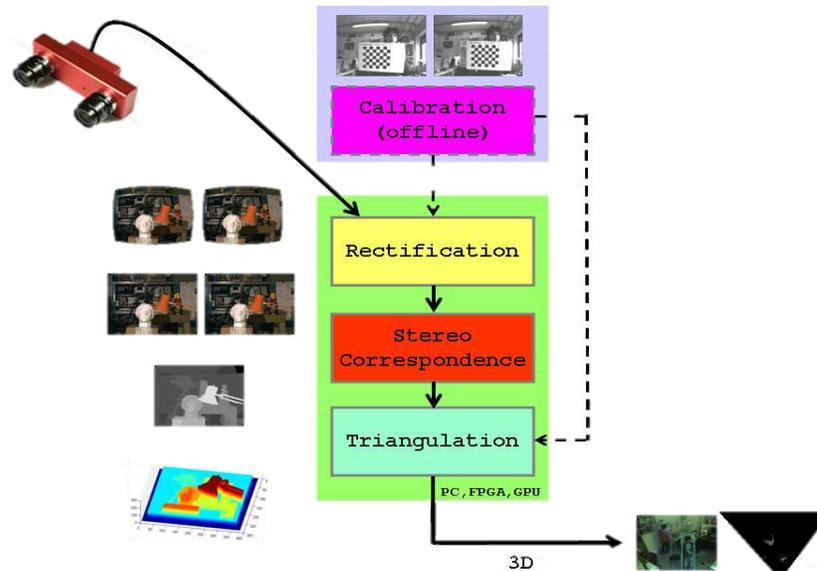


Figura 2.6. Diagrama representativo del proceso informático para aplicar visión estereoscópica.

La visión estereoscópica implica la extracción de información de profundidad desde dos vistas diferentes de una escena obtenida por dos cámaras diferentes. Después de la calibración y luego de la rectificación (que proyecta las imágenes a un plano común) se pueden encontrar los puntos de imagen correspondientes minimizando la función de costo, que en el caso más simple resume las diferencias en las intensidades de los píxeles en la vecindad de dos puntos analizados. Entonces, al comparar estas dos imágenes, se puede calcular el mapa de disparidad que, para este par estereofónico, mantiene diferencias en las coordenadas horizontales de los puntos correspondientes. Los valores en el mapa de disparidad son inversamente proporcionales a la profundidad de la escena. La visión estéreo es similar a la visión binocular humana. Los objetos que están más cerca de nuestros ojos tienen un desplazamiento relativo más grande que los más alejados.

### 2.3.AMPLIFICACIÓN DE POTENCIA

En esta sección se describen las herramientas necesarias para amplificar la potencia de una señal.

### 2.3.1. Modulación por ancho de pulso

Por lo general, todo sistema que procesa información binaria para controlar un proceso analógico requiere una etapa de entrada analógico – digital y una etapa de salida digital–analógica (convertidores ADC y DAC). Para reducir costos en los diseños que no requieren alta resolución en la etapa de salida, es posible sustituir el DAC por un algoritmo de Modulación por Ancho de Pulsos (PWM - Pulse-Width Modulation). Una unidad PWM permite asignar cierta duración de tiempo en alto o en bajo a un dato digital de n bits que se considera salida de la etapa procesadora. Lo anterior se logra conectando un contador y un circuito comparador, como se puede observar en la Figura 2.7.

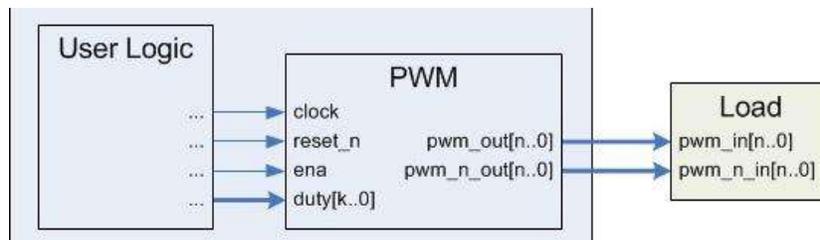


Figura 2.7. Diagrama representativo de la modulación por ancho de pulso.

El comparador determinará si el dato aplicado a la entrada de la unidad es igual al valor binario del contador que cambia constantemente. El tiempo que durará la señal en alto depende de la cantidad de pulsos de reloj que se apliquen hasta que el contador presente un dato binario mayor o igual al de la entrada. A la salida de la unidad PWM es necesario conectar un filtro RC (Pasa Bajas) para determinar el nivel analógico propuesto por el filtro. El periodo completo de un ciclo PWM es igual al producto del periodo del reloj de la señal de referencia (reloj del sistema) con  $2^n$ , donde n es el número de bits del contador propuesto, esto se muestra en la Ecuación 2.2.

$$T_{PWM} = T_{Reloj} * 2^n \quad (2.2)$$

Es posible advertir que si el contador es de 4 bits, se tendrá que  $TPWM = 16 \cdot T_{Reloj}$ , por lo que, al aumentar el número de bits del contador para mejorar la resolución, el periodo PWM tenderá a hacerse más grande reduciendo drásticamente la frecuencia de salida. Lo anterior es sumamente importante cuando se desee realizar una aplicación que sí requiera sincronizar la señal analógica para controlar un sistema. Para los propósitos del diseño planteado, considérese una frecuencia de referencia de 60 Hz, con un contador de 4 bits, y sustituyendo en la Ecuación 2.2, se obtiene una frecuencia PWM de 3.75 Hz, equivalente a un TPWM de 0.26 s. La relación de los valores del filtro está dada por:

$$RC = \frac{1}{2 * \pi * F_{PWM}} \quad (2.3)$$

Por lo que eligiendo  $C = 1 \mu F$ , se obtiene un valor para R de aproximadamente 42 K $\Omega$  (es posible ajustar a un valor comercial más exacto). En nuestro caso particular, se diseñará una unidad PWM de frecuencia baja por lo que no es necesario considerar aspectos de diseño más formales en el filtro. En diseños formales se recomienda estimar una frecuencia de corte menor a la frecuencia PWM y no es común considerarlas iguales, tal y como se hizo en la aproximación anterior. En tal situación será necesario sustituir la frecuencia PWM por la frecuencia de corte en la Ecuación 2.3 o tal vez, hasta considerar un filtro de mayor orden que permita una mejor estabilidad de la señal.

### 2.3.2. Amplificación de señales con transistor.

La función amplificadora consiste en elevar el nivel de una señal eléctrica que contiene una determinada información, como se puede ver en la Figura 2.8.

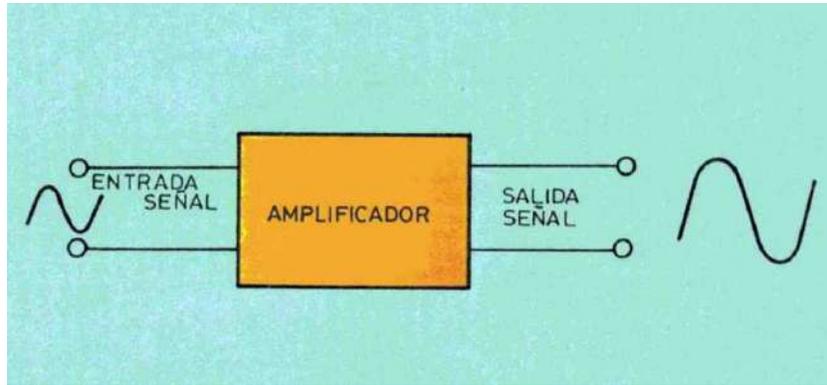
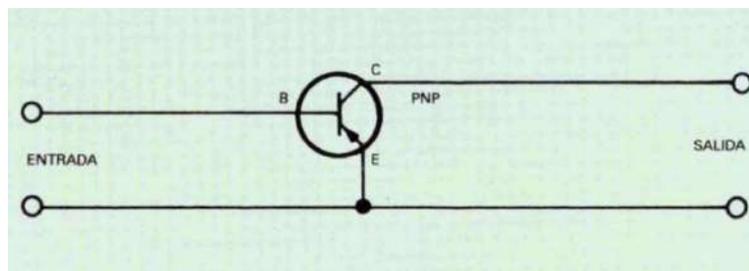
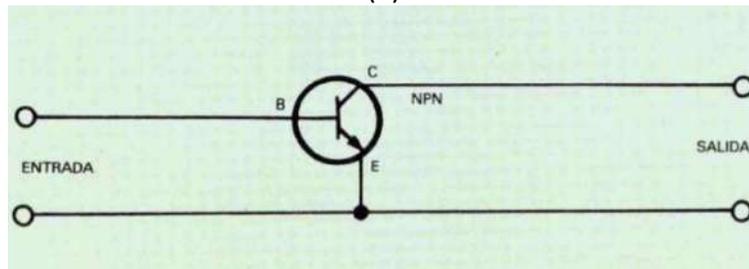


Figura 2.8. Representación gráfica de la función de amplificación.

Esta señal en forma de una tensión y una corriente es aplicada a la entrada del elemento amplificador, originándose una señal de salida conteniendo la misma información, pero con un nivel de tensión y corriente, más elevado. El transistor sea PNP (Figura 2.9 a) ó NPN (Figura 2.9 b)) es capaz de amplificar corriente, es decir, que a una determinada intensidad aplicada en uno de sus terminales de entrada (emisor o base generalmente) responde con una corriente mayor en el de salida (colector). A través de esta forma de trabajo se puede obtener otras amplificaciones como son la de tensión y la potencia.



(a)



(b)

Figura 2.9. a) Diagrama del transistor PNP, b) Diagrama del transistor NPN.

## 2.4.DESCRIPCIÓN DE HARDWARE

En esta sección se describen las herramientas utilizadas pertenecientes al hardware.

### 2.4.1. Sensor

Es un dispositivo que, a partir de la energía del medio donde se mide, da una señal de salida traducible que es función de la variable de medida. La ampliación de los sentidos para adquirir un conocimiento de cantidades físicas que, por su naturaleza o tamaño, no pueden ser percibidas directamente por los sentidos Rincón (2010).

### 2.4.2. Sensor de proximidad

El sensor de proximidad es un dispositivo que detecta objetos o señales que se encuentran cerca del elemento. Existen varios tipos de sensores de proximidad según el principio físico que utilizan. Los más comunes son los interruptores de posición, los detectores capacitivos, los inductivos y los fotoeléctricos, como el de infrarrojos.

### 2.4.3. Sensor de proximidad de ultrasonido

Los ultrasonidos son una radiación mecánica de frecuencia superior a los audibles (20Khz). Toda radiación al incidir sobre un objeto, en parte se refleja, en parte se transmite y en parte es absorbida. Si además hay un movimiento relativo entre la fuente de radiación y el reflector, se produce un cambio de frecuencia de la radiación (efecto Doppler)

Todas estas propiedades de la interacción de una radiación con un objeto han sido aplicadas en mayor o menor grado a la medida de diversas magnitudes físicas. El poder de penetración de la radiación permite que muchas de estas aplicaciones sean totalmente no invasivas, es decir, que no acceda al interior del recinto donde se producen los cambios que se desean detectar Rincón (2010). En

función del tiempo que tarda el sonido en rebotar y volver, se calcula la distancia a la que se encuentra dicho objeto.

#### 2.4.4. Transductor

Se denomina transductor, en general, a todo dispositivo que convierte una señal de una forma física en una señal correspondiente, pero de otra forma física distinta, por tanto, es un dispositivo que convierte un tipo de energía en otro Rincón (2010).

#### 2.4.5. Actuadores de vibración mecánica

Se usará un motor de vibración (Figura 2.10), del tipo motor-moneda 3.0V, aplicables para el dispositivo usable debido a que cumple con los requerimientos de seguridad para poder ser empleados en una persona.



Figura 2.10. Motor de vibración mecánica GlobalSources (2016).

Las especificaciones del motor de vibración se muestran en la Tabla 2.1.

Voltaje de funcionamiento:	2.7 V – 3.3 V DC
----------------------------	------------------

Voltaje clasificado:	3.0 V DC
Voltaje el comenzar:	2.3V DC
Corriente clasificada:	100mA (máximo)
Velocidad clasificada:	12000±3000rpm
Comenzar la corriente:	120mA (máximo)
Resistencia de aislamiento:	10MΩ (mínimo)
Dimensiones principales:	
Diámetro:	8.0mm
Grueso:	2.7mm

Tabla 2.1. Especificaciones de los moto-vibradores empleados.

Usos: dispositivo usable, dispositivo muñeca-basado para el cuidado de la salud, aparato médico eléctrico y otros productos de la vibración.

### 3. DESARROLLO DE UN SISTEMA DE VISIÓN ESTEREOSCÓPICA

En este capítulo se describe de forma detallada la metodología seguida para llevar a cabo este proyecto de tesis dando prioridad a la parte del desarrollo de software.

La metodología general se presenta en el diagrama de la Figura 3.1.

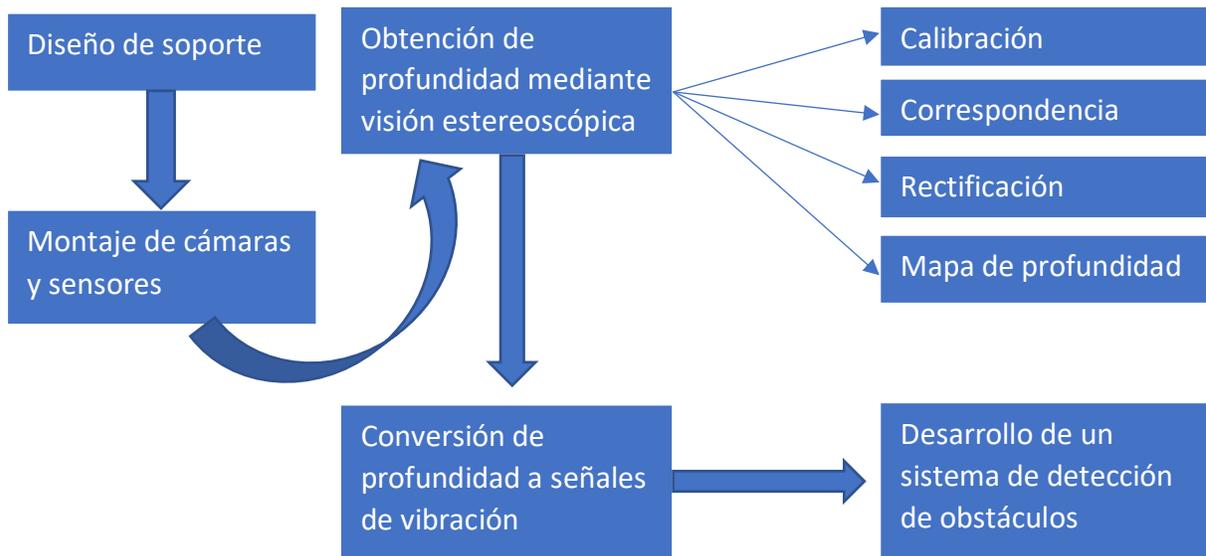


Figura 3.1. Diagrama general de la metodología seguida para el desarrollo del presente proyecto de tesis.

En el área de visión artificial y procesamiento de imágenes la visión estereoscópica es la extracción de información de tres dimensiones a partir de imágenes digitales. Se hace comparando la información de una escena desde dos puntos de vista. Es similar al proceso biológico de estereopsis.

En general, se utilizan dos cámaras desplazadas horizontalmente una de la otra, la profundidad puede ser obtenida comparando ambas imágenes en la forma de disparidades que son inversamente proporcionales a la diferencia en la distancia

a los objetos. Para su implementación se requieren tres pasos de procesamiento los cuales se muestran en el diagrama de la Figura 3.2.

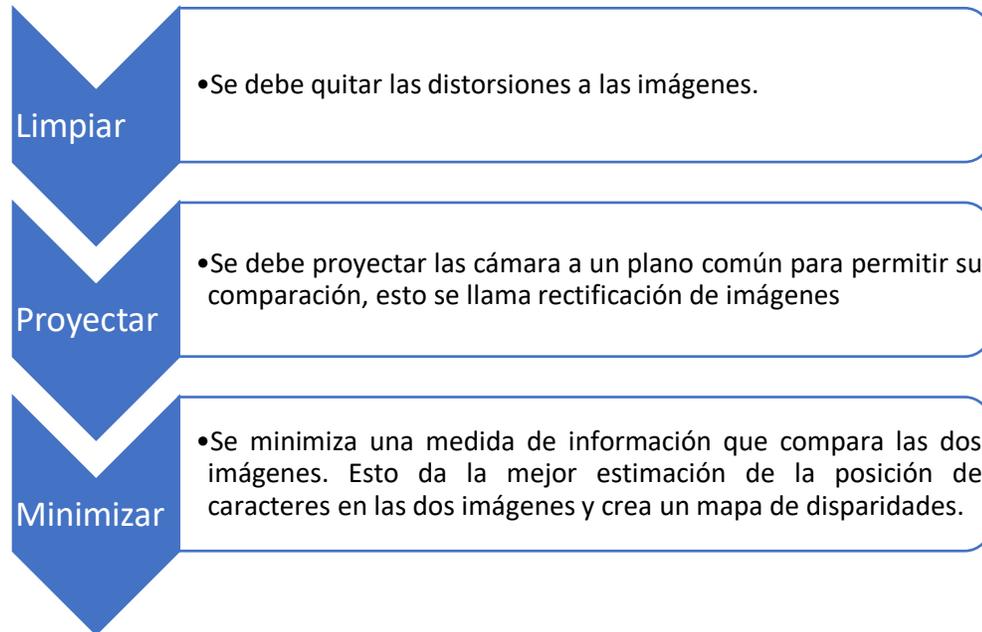


Figura 3.2. Diagrama del procesamiento de imágenes para aplicar visión estereoscópica.

Explicando esto con más detalle, un pixel almacena color en una posición dicha posición se puede identificar por una coordenada en el arreglo de pixeles (x, y) y la profundidad al pixel z.

### 3.1.DISEÑO DE SOPORTE DE CÁMARAS

Es necesario diseñar una base que sostenga dos cámaras, de las cuales se obtienen las imágenes en las que se implementan los algoritmos de visión estereoscópica, para esto se utiliza una herramienta de software de diseño. Este diseño debe cumplir con las medidas y forma adecuadas para colocar las cámaras a la distancia y ángulo adecuados, además, debe mantener la forma de la cabeza de una persona porque el dispositivo es montado a la altura de la frente. En la Figura 3.3 se muestra un diagrama de secuencia del diseño de la pieza.

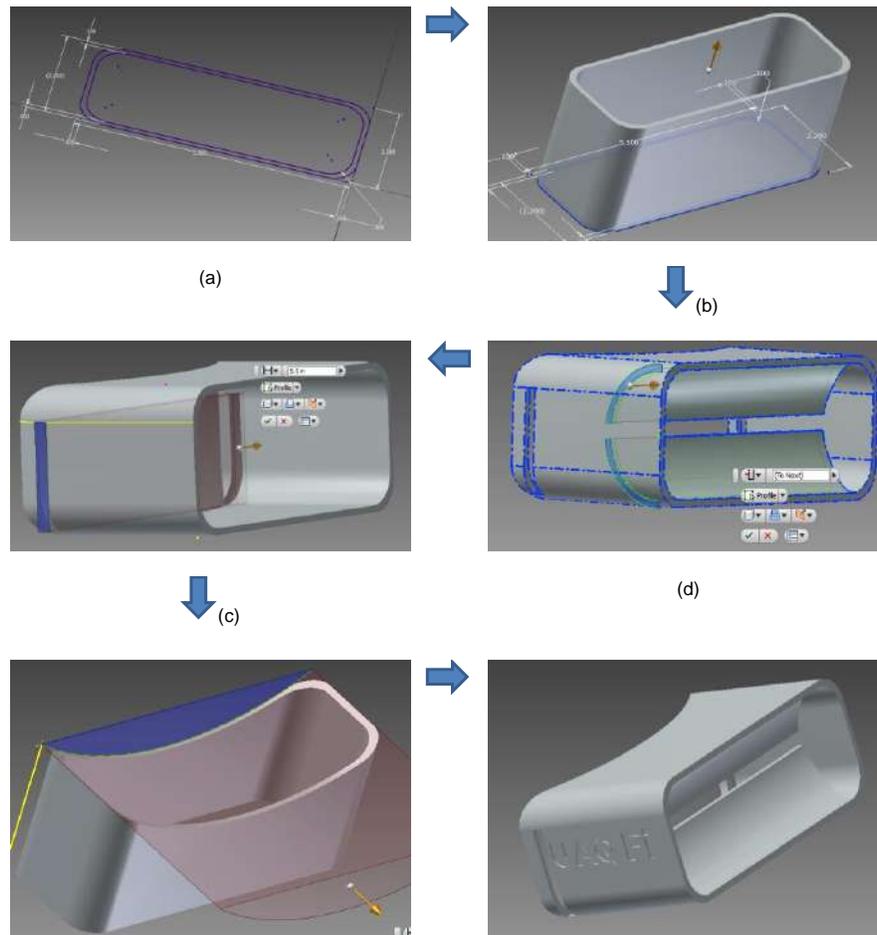


Figura 3.3 Diseño de base para sostener las cámaras

Para el diseño de esta pieza se toman en cuenta varios factores importantes debido a que es un dispositivo portable. Debe ser lo más ligero posible y al mismo tiempo lo suficientemente resistente para soportar dos cámaras, un procesador y una batería, todo esto sin dejar de lado que debe tener un diseño que permita ser colocado en la cabeza de una persona a la altura de la frente. Es por eso que se tomó como referencia el diseño de los binoculares de realidad virtual (VR) convencional, como se puede observar en la Figura 3.3 (a). La longitud frontal que se define está basada en el tamaño de las cámaras y de la batería que estarán colocadas dentro de la base, Figura 3.3 (b). Es necesario dejar dos ranuras laterales para colocar la banda de sujeción que se pone en la cabeza de la persona, Figura 3.3 (c). En la Figura 3.3 (d) se observa que se realizó un extruido en forma de medio

círculo al frente de la base, esto es para colocar las dos cámaras que se están usando para obtener las imágenes de entrada en el programa, de tal forma que sea posible trasladarlas horizontalmente a derecha e izquierda para ajustar la distancia entre cámaras correctamente. Por último, se realiza un corte en la parte trasera de la base en forma de arco para que se adapte a la forma de la frente de una persona, Figura 3.3 (e).

### 3.2.AJUSTE DE DISTANCIA ENTRE CÁMARAS

En base a la teoría revisada sobre visión estereoscópica se sabe que la distancia entre las dos cámaras depende del tamaño del área en la que se desee trabajar, en este caso esta área se define de 2 m de ancho por 6 m de largo, con estos valores se obtiene una distancia entre cámaras de 10 cm y así se puede aplicar la Ecuación 3.1 para obtener la matriz de disparidad como se muestra en la Figura 3.5.

En el diagrama de la Figura 3.5 el punto X es transmitido a través de los puntos de entrada de las cámaras en O y O' hacia las imágenes en x y x'.

$$disparity = x - x' = \frac{Bf}{Z} \quad (3.1)$$

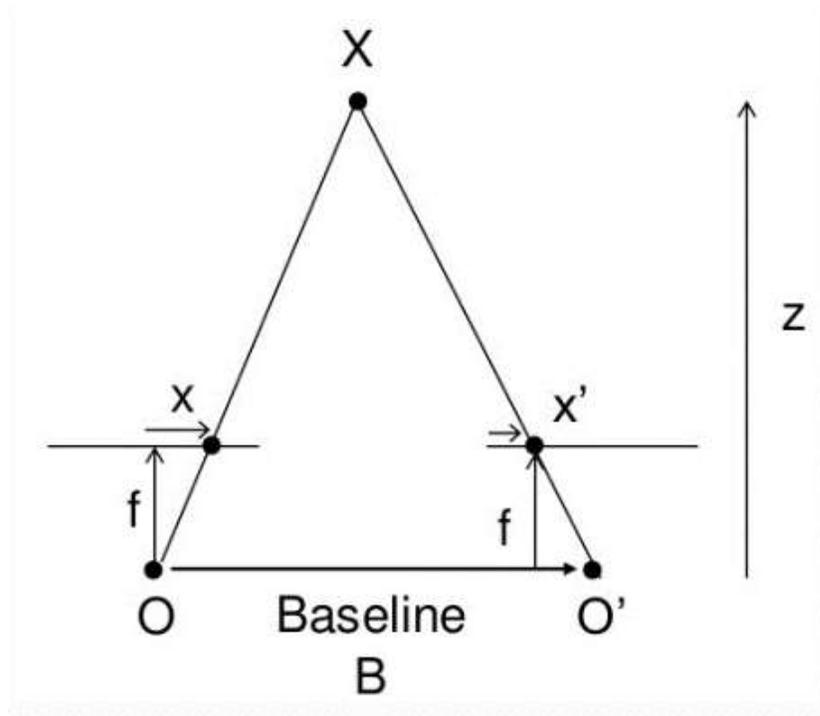


Figura 3.5. Diagrama representativo de la ecuación de disparidad.

En el apéndice 2 se muestra parte del código de programación aplicado para llevar acabo este proceso:

Una vez definida la distancia adecuada entre las cámaras se procede a hacer el montaje de las mismas como se muestra en la Figura 3.6, respetando un ángulo de operación entre ellas de  $5^\circ$  para que se realice la calibración y comenzar a realizar las primeras capturas. (Baseline: Línea donde coinciden los 2 ejes horizontales de los marcos coordenados)

### 3.3.ADQUISICIÓN DE IMÁGENES

Es necesario que el sistema implementado sea capaz de leer ambas cámaras y mostrar las imágenes que se estén adquiriendo tanto de la cámara izquierda como de la derecha (Left y Right, respectivamente). Entonces, se

almacenan las imágenes en una carpeta con indicativo si la imagen es de la cámara izquierda o de la derecha, estas imágenes se almacenan tanto a color como a escala de grises.

Para no sacrificar el rendimiento del procesador las imágenes se guardan a una resolución de 1280x1024 pixeles, pero se muestran a una resolución de 640x480 pixeles. Las imágenes se almacenan en carpetas diferentes (Left y Right) para facilitar su uso en la aplicación de calibración del sistema desarrollado en Python.

### 3.4.CALIBRACIÓN DE CÁMARAS

Existen dos formas diferentes de calibrar las cámaras para aplicar visión estereoscópica.

- Extrínseca

Busca relación entre sistemas coordenados del entorno y de la cámara, se preservan unidades de longitud y entrega como salida dos matrices una de rotación y una de traslación.

- Intrínseca

Busca la relación entre el sistema coordenado de la cámara y de la imagen, realiza una conversión entre unidades de longitud y de pixeles y entrega a la salida la razón del centro de la imagen, sesgo y distancias focales.

Para llevar a cabo la calibración se sigue el siguiente procedimiento.

Es necesario tomar 19 imágenes de un tablero de ajedrez con cada una de las cámaras para implementar los algoritmos de calibración predefinidos en las librerías de OpenCV como se muestra en la Figura 3.7.

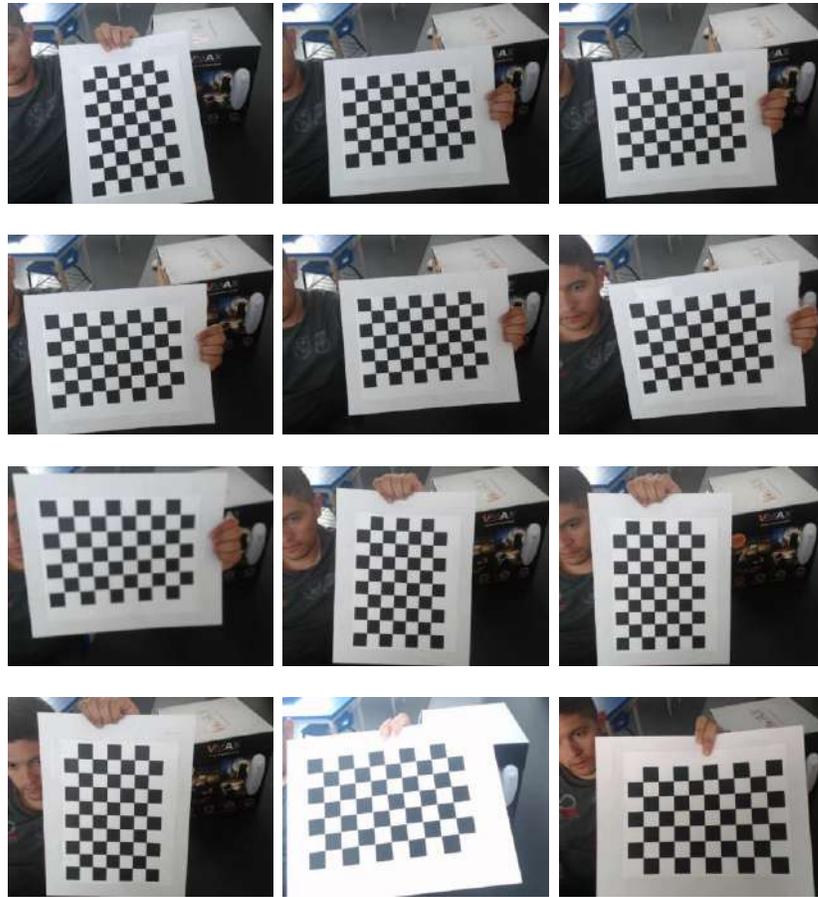


Figura 3.7. Imágenes capturadas para la calibración de las cámaras.

La carpeta en la que se guardan estas imágenes de estar en la ubicación que el archivo en el que se ejecuta el código de calibración y la ruta no debe ser muy larga. Mientras se ejecuta, la aplicación pedirá el nombre de la carpeta con el set a calibrar.

Dentro de la carpeta se creará un folder llamado “*CalibResult*” donde se guardan las imágenes rectificadas. En el folder raíz se almacenan los archivos *CameraMatrix*, *distCoef*, *rvecs* y *tvecs* como archivos *.npz* y como archivos de texto *.txt*. Entrega un archivo más con el error en la estimación y se almacena como archivo de texto. Las imágenes de entrada originales son corregidas para contrastarlas entre sí y ver el efecto de la calibración. Se muestran algunos ejemplos en la Figura 3.7.

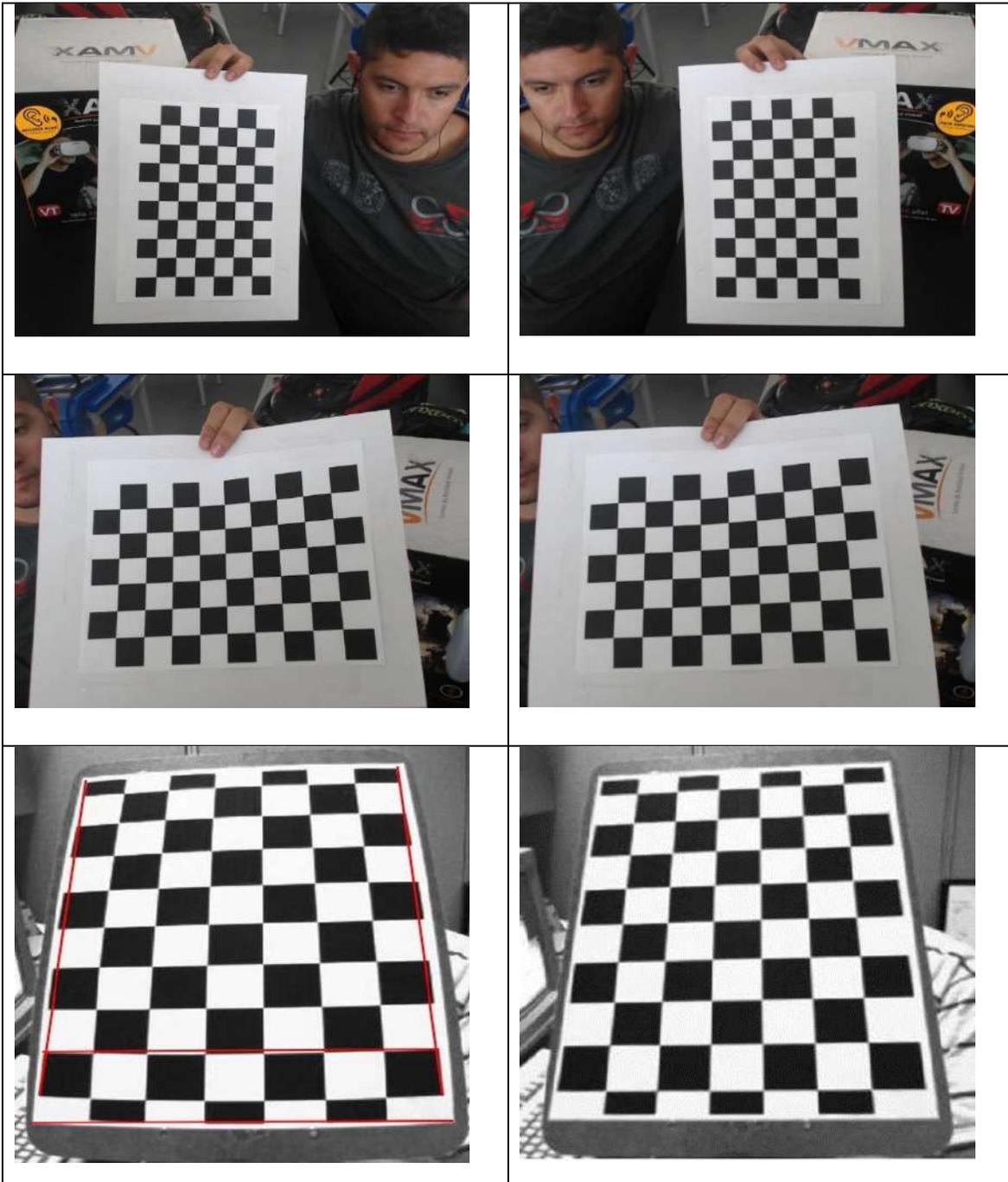


Figura 3.8. Figuras corregidas con la calibración.

Una vez que se obtienen los archivos de calibración de las imágenes y los resultados esperados es posible tomar las primeras capturas para comenzar a realizar las operaciones de correspondencia y obtención de profundidad.

## 4. CONVERSIÓN DE INFORMACIÓN TRIDIMENSIONAL A DIAGRAMA DE FRECUENCIA

Es necesario ingeniar una manera de traducir la información en tres dimensiones al diagrama de frecuencia para poder transmitir esta información a una persona mediante actuadores de vibración mecánica. Por esta razón, se desarrollaron fórmulas para traducir información del espacio al diagrama de frecuencia, como se muestran en la Tabla 4.1.

Fórmulas desarrolladas	
Función de frecuencia en el eje X	$F_{zx}(K) = F_{ix}(KT) * F_{max} + F_{min}$ <p>Donde:  <math>K=1,2,3,\dots,N</math>  <math>F_{max} \propto WS_z</math>  <math>F_{min} \propto F_{zx(0)}</math></p>
Función de frecuencia en el eje Y	$F_{zy}(K) = F_{iy}(KT) * F_{max} + F_{min}$ <p>Donde:  <math>K=1,2,3,\dots,N</math>  <math>F_{max} \propto WS_z</math>  <math>F_{min} \propto F_{zx(0)}</math></p>
Número total de datos	$N = \frac{t}{T}$

Tabla 4.1. Fórmulas de traducción de información.

Definición de variables		Unidades
$t$	Tiempo de vibración	Segundos
$F_{min}$	Frecuencia mínima de vibración	Hertz
$F_{max}$	Frecuencia máxima de vibración	Hertz
$WS_x$	Espacio de trabajo sobre el eje X	Metros
$WS_y$	Espacio de trabajo sobre el eje Y	Metros
$WS_z$	Espacio de trabajo sobre el eje Z	Metros
$f_{ix}(t)$	Función que representa la proyección del espacio de trabajo en el plano "XZ"	Dominio: $(0-\infty)$ Co-dominio: $(0-1)$
$f_{iy}(t)$	Función que representa la proyección del espacio de trabajo en el plano "YZ"	Dominio: $(0-\infty)$ Co-dominio: $(0-1)$
$T$	Periodo de muestreo	Milisegundos
$N$	Numero total de datos	-
$K$	Contador	De $(0$ a $N)$

Tabla 4.2 representación de las variables en las fórmulas desarrolladas.

También se realizaron pruebas para comprobar el correcto comportamiento de las señales con las fórmulas desarrolladas una de estas pruebas fue la que se muestra en la Tabla 4.2 y en la Figura 4.3.

Con los valores de entrada presentados en la Tabla 4.2 se procede a graficar la entrada y la salida en los algoritmos de traducción de información y se obtienen las gráficas mostradas en la Figura 4.1.

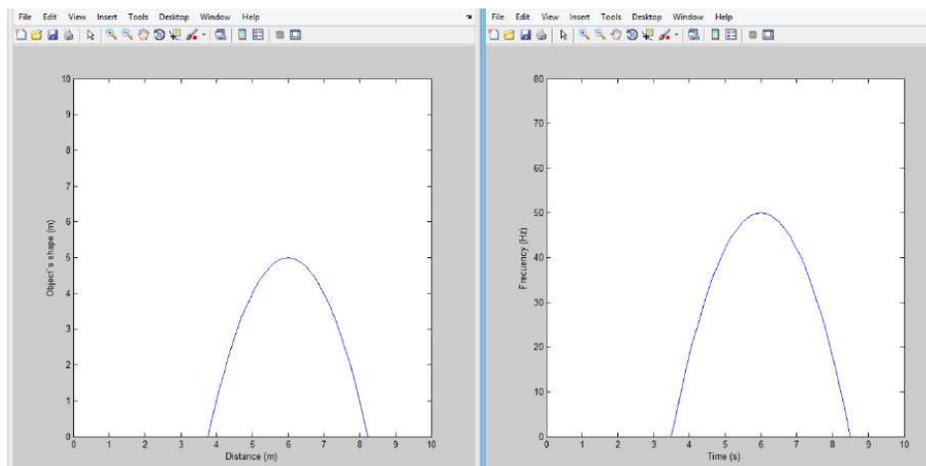


Figura 4.1. Gráficas de las funciones traducidas del espacio al diagrama de frecuencia.

En el lado izquierdo de la Figura 4.3 se muestra la función de entrada que muestra la forma que tiene un objeto y en el lado derecho se muestra la gráfica de la función resultante al pasar por las operaciones de las fórmulas desarrolladas. Con esto se comprobó que las fórmulas están correctas dada la similitud entre las dos señales, pero en diferente diagrama.

## 5. DESARROLLO DE UN SISTEMA PARA DETECCIÓN DE OBSTÁCULOS APLICANDO VISIÓN ARTIFICIAL

En esta sección se describe el acoplamiento y la unión de las diferentes etapas en el desarrollo del presente proyecto de Tesis, desde su implementación hasta las pruebas realizadas a nivel software y hardware y la discusión de los resultados.

Con el diseño de la pieza listo, se imprime en una impresora 3D con material del tipo PLA color blanco como se muestra en la Figura 5.1.



Figura 5.1. Base para sostener las cámaras a la distancia adecuada.

Entonces se realiza el ensamble de las dos cámaras, para ello se necesita calcular el ángulo y la distancia entre ellas.



Figura 5.2 Montaje de cámaras a la distancia adecuada.

Se procede a tomar diferentes capturas del espacio de trabajo definido para la obtención de profundidad como se muestra en la Figura 5.3.





Figura 5.3. Captura de imágenes para obtención de profundidad.

Con las imágenes almacenadas se emplea un código que permite encontrar puntos similares entre las dos imágenes (Left y Right) y de la misma forma nos entrega una matriz llamada Matriz fundamental que nos dice como está una imagen con respecto a la otra.

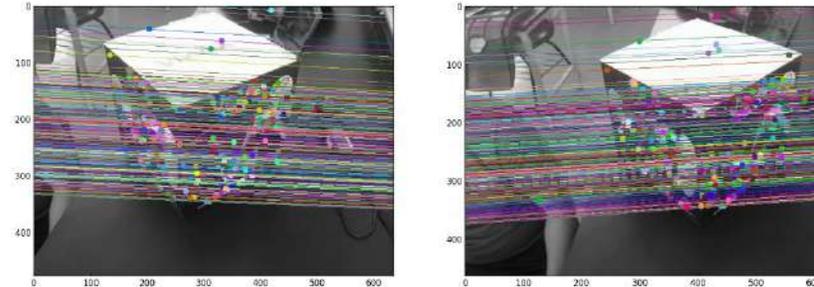


Figura 5.4. Puntos de correspondencia en ambas imágenes.

Del procedimiento anterior se obtiene un matriz fundamental como se muestra a continuación.

$$\begin{bmatrix} [-6.09786898e - 07 & - 5.79285937e - 06 & 4.79779533e - 04] \\ [ 5.44162590e - 06 & 1.48324184e - 06 & 1.66098033e - 02] \\ [-2.08378102e - 03 & - 1.82584761e - 02 & 1.00000000e + 00] \end{bmatrix}$$

Esta matriz nos da información detallada sobre la rotación y traslación que hay entre las dos imágenes. Entonces nos entrega una última imagen rectificadas en la que se alinean los puntos encontrados como se muestra en la Figura 5.5.



Figura 5.5. Imágenes rectificadas con puntos de coincidencia corregidos.

### 5.1.IMPLEMENTACIÓN DE ACTUADORES DE VIBRACIÓN

Con el objetivo de transmitir la información tridimensional al usuario del dispositivo se diseña y se fabrica un guante que contenga actuadores de vibración del tipo motor de vibración. Estos actuadores se colocan en la parte superior de los nudillos de la mano derecha para que se tenga una mayor sensibilidad y al mismo tiempo, evitar que al usuario le afecte en sus actividades cotidianas y que no le estorbe.



Figura 5.6. Colocación de actuadores de vibración.

## 5.2. PRUEBAS DEL DISPOSITIVO Y DISCUSIÓN DE RESULTADOS

Se realizan diferentes pruebas en el dispositivo desarrollado, desde las pruebas en los actuadores de vibración para asegurar que la frecuencia de vibración sea directamente proporcional al ancho de pulso enviado desde el procesador.

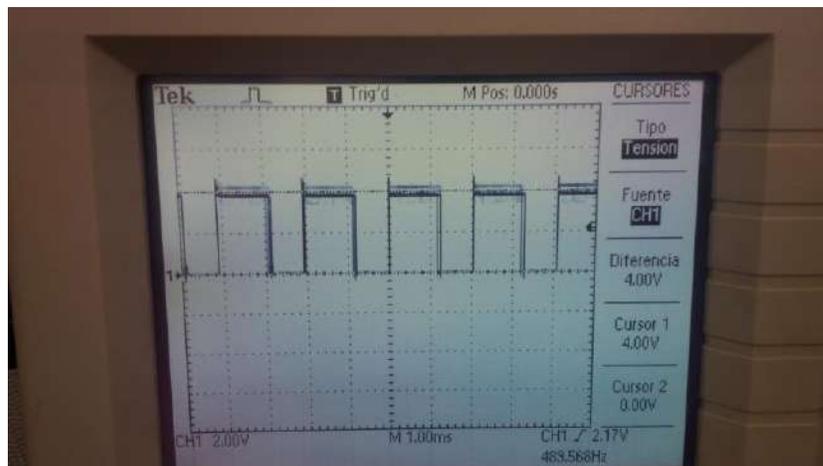


Figura 5.7. Comprobación de linealidad en la frecuencia de vibración.

Dentro de las pruebas que se consideran más críticas están las hechas en la etapa de calibración en la que se obtuvieron los siguientes resultados:

<b>Error total</b>	7.7%	
<b>Matriz de traslación</b>	6.642347822335568708e+02 0.000000000000000000e+00 3.139105104183068420e+02 0.000000000000000000e+00 6.659787720273545801e+02 2.240475804830724655e+02 0.000000000000000000e+00 0.000000000000000000e+00 1.000000000000000000e+00	
<b>Vectores R</b>	3.103696119736828685e+00 -1.760337476672478307e-01 2.923701363828284716e+00	7.358118505535603260e-02
<b>Vectores T</b>	4.593512778645949091e+00 2.431891307111194322e+01 2.581029083832780291e+00 2.456475006829727548e+01	6.198121480846672249e+00 6.311774573702823687e+00
<b>Valores de distorsión</b>	3.236203621509662343e-01 2.731580943327580058e-02 1.963964784876178760e+00	-1.003300804310552685e+00 - -3.687010115567434642e-03

Tabla 5.1. Resultados de la calibración de las cámaras.

También se realizaron pruebas para comprobar el correcto comportamiento de las señales con el modelo propuesto, una de estas pruebas se muestra en la Tabla 5.2 y en la Figura 5.8.

Valores de entrada	
%VALORES DE ENTRADA t=0.1:0.1:10;	Segundos
Fmin=10; %Hz Fmax=80; %Hz	Hertz

$WSx=10;$ %m $WSy=10;$ %m $WSz=10;$ %m	<p style="text-align: center;">Metros</p>
$f_{ix}=-\frac{1}{2}(t-4)^2+5;$ %m $f_{iy}=-\frac{1}{2}(t-6)^2+5;$ %m	<p style="text-align: center;">Metros</p>

Tabla 5.2. Valores de entrada en la comprobación de las fórmulas desarrolladas.

Con los valores de entrada presentados en la Tabla 5.2 se procede a graficar la entrada y la salida mediante el modelo propuesto con lo que se obtienen las gráficas mostradas en la Figura 5.3.

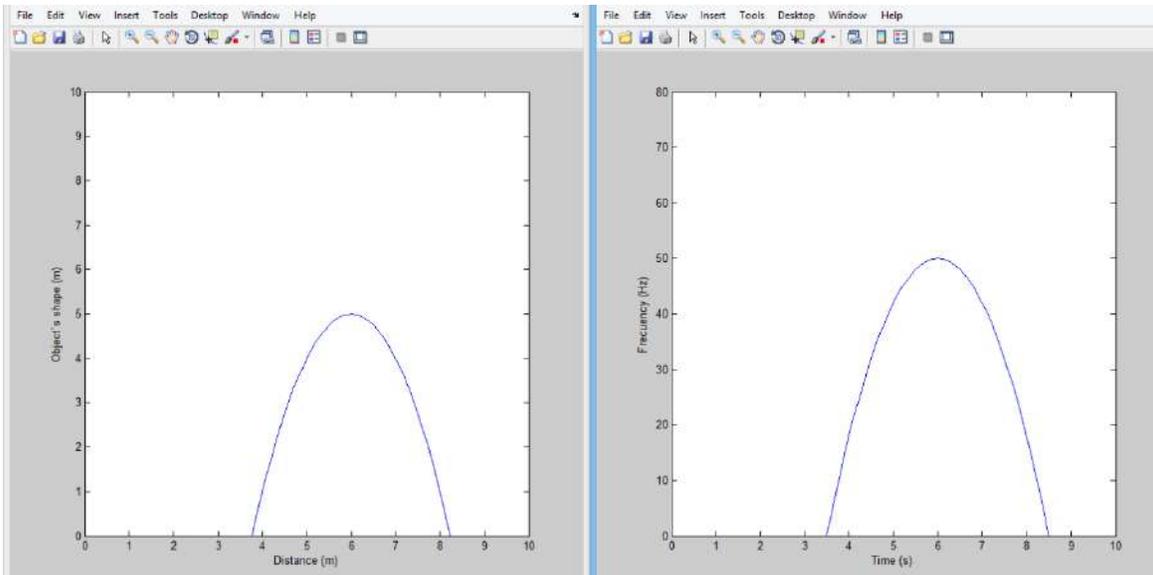


Figura 5.8. Gráficas de las funciones traducidas del espacio al diagrama de frecuencia.

En el lado izquierdo de la Figura 5.3 se muestra la función de entrada que muestra la forma que tiene un objeto y en el lado derecho se muestra la gráfica de la función resultante al pasar por las operaciones de las fórmulas desarrolladas. Con

esto se comprobó que las fórmulas están correctas dada la similitud entre las dos señales, pero en diferente diagrama.

## 6. CONCLUSIONES

Se desarrollaron fórmulas para traducir información del espacio al diagrama de frecuencia, éstas están representadas en la Tabla 6.1.

Fórmulas desarrolladas	
Función de frecuencia en el eje X	$F_{zx}(K) = F_{ix}(KT) * F_{max} + F_{min}$ <p>Donde:  <math>K=1,2,3,\dots,N</math>  <math>F_{max} \propto WS_x</math>  <math>F_{min} \propto F_{zx}(0)</math></p>
Función de frecuencia en el eje Y	$F_{zy}(K) = F_{iy}(KT) * F_{max} + F_{min}$ <p>Donde:  <math>K=1,2,3,\dots,N</math>  <math>F_{max} \propto WS_y</math>  <math>F_{min} \propto F_{zy}(0)</math></p>
Número total de datos	$N = \frac{t}{T}$

Tabla 6.1. Fórmulas de traducción de información.

	Definición de variables	Unidades
$t$	Tiempo de vibración	Segundos
$F_{min}$	Frecuencia mínima de vibración	Hertz
$F_{max}$	Frecuencia máxima de vibración	Hertz
$WS_x$	Espacio de trabajo sobre el eje X	Metros
$WS_y$	Espacio de trabajo sobre el eje Y	Metros
$WS_z$	Espacio de trabajo sobre el eje Z	Metros
$f_{ix}(t)$	Función que representa la proyección del espacio de trabajo en el plano "XZ"	Dominio: $(0-\infty)$ Co-dominio: $(0-1)$
$f_{iy}(t)$	Función que representa la proyección del espacio de trabajo en el plano "YZ"	Dominio: $(0-\infty)$ Co-dominio: $(0-1)$
$T$	Periodo de muestreo	Milisegundos
$N$	Numero total de datos	-
$K$	Contador	De $(0$ a $N)$

Tabla 6.2 representación de las variables en las fórmulas desarrolladas.

Se cuenta con un dispositivo que obtiene un mapa de profundidad en un espacio determinado y convierte la información en una combinación de señales de vibración que se implementan con motores de vibración mecánica.

En base a la literatura revisada se sabe que los dispositivos con un funcionamiento similar presentes en el mercado actualmente tienen un precio aproximado de \$300,000 MXN mientras que el prototipo de este proyecto de Tesis tiene un costo aproximado de \$10,000 MXN en cuestión del material, es decir, sin tomar en cuenta el costo por el diseño y el desarrollo de software.

Como trabajo futuro se pretende implementar el prototipo en un sistema embebido y lograr uno de los principales objetivos de este proyecto que es hacerlo portátil para una persona y fácil de usar, esto aún no se logra debido a que los procesadores que nos permiten tener un sistema embebido no son lo suficientemente robustos para llevar a cabo el procesamiento de la información.

Si bien, ya se tiene el sistema funcionando en la obtención de información así como la traducción y transmisión de la misma, aún falta montar el sistema en una persona y comprobar que la persona sea capaz de interpretar las señales recibidas.

## 7. REFERENCIAS

Zereen, A. N., & Corraya, S. (2016, December). Detecting real time object along with the moving direction for visually impaired people. In Electrical, Computer & Telecommunication Engineering(ICECTE), International Conference on (1-4).

Rashid, H., Al-Mamun, A. R., Robin, M. S. R., Ahasan, M., & Reza, S. T. (2016, December). Bilingual wearable assistive technology for visually impaired persons. In Medical Engineering, Health Informatics and Technology (MediTec), 2016 International Conference on (1-6).

Everding, L., Walger, L., Ghaderi, V. S., & Conradt, J. (2016, September). A mobility device for the blind with improved vertical resolution using dynamic vision sensors. In e-Health Networking, Applications and Services (Healthcom), 2016 IEEE 18th International Conference on (1-5).

Hejun, W., Peksi, S., & Seng, G. W. (2015, December). An affordable and attachable electronic device for the blind. In Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2015 Asia-Pacific (559-562).

Ghaderi, V. S., Mulas, M., Pereira, V. F. S., Everding, L., Weikersdorfer, D., & Conradt, J. (2015, August). A wearable mobility device for the blind using retina-inspired dynamic vision sensors. In Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE (3371-3374)

bin Mohamed Kassim, A., Yasuno, T., Jaafar, H. I., Aras, M. S. M., & Abas, N. (2015, July). Performance analysis of wireless warning device for upper body level of deaf-blind person. In Society of Instrument and Control Engineers of Japan (SICE), 2015 54th Annual Conference of the (252-257).

Ramirez-Garibay, F., Olivarria, C. M., Aguilera, A. F. E., & Huegel, J. C. (2014, October). MyVox—Device for the communication between people: blind,

deaf, deaf-blind and unimpaired. In Global Humanitarian Technology Conference (GHTC), 2014 IEEE (506-509).

Raghuvanshi, D. S., Dutta, I., & Vaidya, R. J. (2014, July). Design and analysis of a novel sonarbased obstacle-avoidance system for the visually impaired and unmanned systems. In Embedded Systems (ICES), 2014 International Conference on (238-243).

Patchava, V., Kandala, H. B., & Babu, P. R. (2015, December). A Smart Home Automation technique with Raspberry Pi using IoT. In Smart Sensors and Systems (IC-SSS), International Conference on (1-4).

Yin, J., & Yang, X. (2016, July). 3D facial reconstruction of based on OpenCV and DirectX. In Audio, Language and Image Processing (ICALIP), 2016 International Conference on (341-344).

Sung, H. W., Chang, Y. M., Wang, S. C., & Lee, J. K. (2016, August). OpenCV Optimization on Heterogeneous Multi-core Systems for Gesture Recognition Applications. In Parallel Processing Workshops (ICPPW), 2016 45th International Conference on (59-65).

Hossain, N., Kabir, M. T., Rahman, T. R., Hossen, M. S., & Salauddin, F. (2015, November). A realtime surveillance mini-rover based on OpenCV-Python-JAVA using Raspberry Pi 2. In Control System, Computing and Engineering (ICCSCE), 2015 IEEE International Conference on (476-481).

Mattocchia, S. (2011). Stereo vision: algorithms and applications. University of Bologna, 22.

Rincón Ruiz, J. E. (2010). Diseño y construcción de un dispositivo electrónico para la detección de obstáculos, como ayuda a personas con discapacidad visual.

## APÉNDICE 1. CÓDIGO DE CALIBRACIÓN DE CÁMARAS

```
import numpy as np
import cv2
import glob
import os

directory_name = raw_input("-> Directory name: ")
print directory_name

directoryCalibResult = directory_name + "/calibResult"

if not os.path.exists(directoryCalibResult):
    os.makedirs(directoryCalibResult)

# termination criteria
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 30, 0.001)

# prepare object points, like (0,0,0), (1,0,0), (2,0,0) ....., (6,5,0)
objp = np.zeros((9*6,3), np.float32)
objp[:, :2] = np.mgrid[0:6,0:9].T.reshape(-1,2)

# Arrays to store object points and image points from all the images.
objpoints = [] # 3d point in real world space
imgpoints = [] # 2d points in image plane.

images = glob.glob(directory_name + '/*.png')
print images

for fname in images:
    img = cv2.imread(fname)
    gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

    # Find the chess board corners
    ret, corners = cv2.findChessboardCorners(gray, (6,9),None)

    # If found, add object points, image points (after refining them)
    if ret == True:
        objpoints.append(objp)

        corners2 = cv2.cornerSubPix(gray,corners,(11,11),(-1,-1),criteria)
        imgpoints.append(corners2)

    # Draw and display the corners
```

```

img = cv2.drawChessboardCorners(img, (6,9), corners2,ret)
cv2.imshow('img',img)
cv2.waitKey(500)

cv2.destroyAllWindows()

ret, mtx, dist, rvecs, tvecs = cv2.calibrateCamera(objpoints, imgpoints, gray.shape[:-1], None,
None)

np.savez(directory_name + '/TOF.npz', mtx=mtx, dist=dist, rvecs=rvecs, tvecs=tvecs)
np.savetxt(directory_name + '/mtx.txt', mtx, delimiter=' ') # mtx is an array
np.savetxt(directory_name + '/dist.txt', dist, delimiter=' ') # dist is an array
np.savetxt(directory_name + '/rvecs.txt', rvecs, delimiter=' ') # rvecs is an array
np.savetxt(directory_name + '/tvecs.txt', tvecs, delimiter=' ') # dist is an array

# Error estimation
mean_error = 0
for i in xrange(len(objpoints)):
    imgpoints2, _ = cv2.projectPoints(objpoints[i], rvecs[i], tvecs[i], mtx, dist)
    error = cv2.norm(imgpoints[i],imgpoints2, cv2.NORM_L2)/len(imgpoints2)
    mean_error += error

print "total error: ", mean_error/len(objpoints)
file = open(directory_name + "/totalError.txt", "w")
file.write(str(mean_error/len(objpoints)))
file.close()

i = 1

for fname in images:
    img = cv2.imread(fname)
    h, w = img.shape[:2]
    newcameramtx, roi=cv2.getOptimalNewCameraMatrix(mtx,dist,(w,h),1,(w,h))

    # undistort
    dst = cv2.undistort(img, mtx, dist, None, newcameramtx)

    # crop the image
    x,y,w,h = roi
    dst = dst[y:y+h, x:x+w]
    cv2.imwrite(directoryCalibResult + '/calibResult_'+ fname[31:],dst)
    i += 1

```

## APÉNDICE 2. CÓDIGO DE MAPA DE PROFUNDIDAD

```
import cv2
import datetime
import os

# Get time
today = datetime.date.today()
time = datetime.datetime.now()

# Directories definition/creation
directory = str(today) + "-" + str(time.hour) + "-" + str(time.minute) + "-" + str(time.second)
directoryGrayImagesL = str(today) + "-" + str(time.hour) + "-" + str(time.minute) + "-" +
str(time.second) + "C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/grayImagesL"
directoryGrayImagesR = str(today) + "-" + str(time.hour) + "-" + str(time.minute) + "-" +
str(time.second) + "C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/grayImagesR"
directoryColorImages = str(today) + "-" + str(time.hour) + "-" + str(time.minute) + "-" +
str(time.second) + "C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/colorImages"

if not os.path.exists(directory):
    os.makedirs(directory)
if not os.path.exists(directoryGrayImagesL):
    os.makedirs(directoryGrayImagesL)
if not os.path.exists(directoryGrayImagesR):
    os.makedirs(directoryGrayImagesR)
if not os.path.exists(directoryColorImages):
    os.makedirs(directoryColorImages)

# Left camera configuration
capL = cv2.VideoCapture(1)
capL.set(3, 1208) # Width
capL.set(4, 1024) # Height

# Right camera configuration
capR = cv2.VideoCapture(0)
capR.set(3, 1208) # Width
capR.set(4, 1024) # Height

counter = 1

while(True):
```

```

# Capture frame-by-frame
retR, frameR = capR.read()
retL, frameL = capL.read()

frameR = cv2.flip(frameR, flipCode=0)

# Our operations on the frame come here
grayR = cv2.cvtColor(frameR, cv2.COLOR_BGR2GRAY)
grayL = cv2.cvtColor(frameL, cv2.COLOR_BGR2GRAY)

# Display the resulting frame
grayRScaled = cv2.resize(grayR, (640, 480))
cv2.imshow('frameR', grayRScaled)
grayLScaled = cv2.resize(grayL, (640, 480))
cv2.imshow('frameL', grayLScaled)

if cv2.waitKey(1) & 0xFF == ord('q'):
    break
elif cv2.waitKey(1) & 0xFF == ord('s'):
    cv2.imwrite(str(directoryColorImages) + 'C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/img' + str(counter) +
'R.png', frameR)
    cv2.imwrite(str(directoryGrayImagesR) + 'C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/img' + str(counter) +
'R.png', grayR)
    cv2.imwrite(str(directoryColorImages) + 'C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/img' + str(counter) +
'L.png', frameL)
    cv2.imwrite(str(directoryGrayImagesL) + 'C:/Users/Salvador Mtz
Cruz/Desktop/TesisMaestria/stereoAcquire/stereoAcquire/2018-05-21-10-11-18/img' + str(counter) +
'L.png', grayL)
    print "Guardada img" + str(counter) + 'R.png'
    print "Guardada img" + str(counter) + 'L.png'
    counter += 1

```