



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERÍA EN ELECTRÓNICA**  
**TELECOMUNICACIONES Y REDES**

**SISTEMA RADAR PARA ANÁLISIS Y RECONSTRUCCIÓN DE**  
**PERFILES TOPOGRÁFICOS A MEDIA ESCALA**

**TESIS DE GRADO**  
**PREVIA A LA OBTENCIÓN DEL TÍTULO DE:**

**INGENIERO EN ELECTRÓNICA TELECOMUNICACIONES Y REDES**

**AUTORES: RENÉ JAVIER ÁLVAREZ GONZÁLEZ**  
**SANDRA ISABEL SHAGÑAY VIMOS**  
**TUTOR: ING. HUGO MORENO AVILES**

**RIOBAMBA – ECUADOR**

**2015**

## **AGRADECIMIENTO**

A Dios por guiarme en mi camino e iluminarme en mis decisiones.

A mis padres por su comprensión, aliento y ayuda durante el transcurso de mis estudios.

A la ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO, a la Facultad de Informática y Electrónica, a la Escuela de Ingeniería Electrónica en Telecomunicaciones y Redes por los conocimientos adquiridos durante mi trayecto universitario.

A nuestro director de Tesis Ing. Hugo Moreno que con su ayuda y guía fue posible culminar este trabajo investigativo.

René Javier Álvarez González

## **AGRADECIMIENTO**

El presente trabajo quiero agradecer a Dios que por su infinita bondad que ha sabido guiarme y permitirme culminar con éxito una más de mis metas.

Un especial reconocimiento a mi compañero incondicional de vida Ángel quien ha sido el principal soporte al darme la fortaleza de continuar con este sueño hoy ya hecho realidad.

A mis Padres, por haberme apoyado incondicionalmente en el transcurso del cumplimiento de mis objetivos que con su ejemplo me instruyeron para continuar con mi vida profesional.

A la ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO por haberme dado la oportunidad de pasar una etapa de mi vida en sus aulas, que mediante sus maestros me inculcaron los conocimientos y la amistad de verdaderos amigos.

También un agradecimiento a nuestro director de Tesis Ing. Hugo Moreno que a través de su experiencia, sus conocimientos y la motivación para lograr terminar esta etapa con éxito.

Isabel

## **DEDICATORIA**

A Dios por haberme dado salud, vida, sabiduría y fortaleza para seguir adelante permitiéndome culminar esta etapa de mi vida.

A mis padres Alfonso Álvarez y Morayma González, gracias a su apoyo incondicional esfuerzo, sacrificio y a su ayuda se logró alcanzar este logro.

Rene Javier Álvarez González

## **DEDICATORIA**

Este trabajo dedico a Dios, que por su infinito amor y bondad me ha permitido fortalecer mi mente y corazón dándome la paciencia necesaria para continuar y no decaer en el intento sobre todo permitirme conocer el amor verdadero.

A mi gran amor Lesly quien en su corta edad supo entenderme al no pasar mayor tiempo a su lado siendo mi principal inspiración y por quien cada día tiene sentido, a ti mi bebida preciosa, mi esperanza, mi alegría, mi vida y la culminación de este proyecto y lo que representa.

A mi Ángel quien supo guiarme para continuar y no desmayar en los problemas que se presentaban, enseñándome a encarar las adversidades sin perder la dignidad ni desfallecer en el intento permitiéndome ser muy feliz.

Isabel

## **FIRMAS DE RESPONSABLES Y NOTA**

| <b>NOMBRE</b>  | <b>FIRMA</b> | <b>FECHA</b> |
|--|--------------|--------------|
| <b>ING. GONZALO SAMANIEGO<br/>DECANO DE LA FACULTAD DE<br/>INFORMÁTICA Y ELECTRÓNICA</b>                             | .....        | .....        |
| <b>ING. FRANKILN MORENO<br/>DIRECTOR DE LA ESCUELA DE<br/>INGENIERIA ELECTRÓNICA,<br/>TELECOMUNICACIONES Y REDES</b> | .....        | .....        |
| <b>ING. HUGO MORENO<br/>DIRECTOR DE TESIS</b>  | .....        | .....        |
| <b>ING. OSWALDO MARTINEZ<br/>MIEMBRO DEL TRIBUNAL</b>  | .....        | .....        |
| <b>COORDINADOR<br/>SISBIB-ESPOCH</b>   | .....        | .....        |

**NOTA DE LA TESIS:** .....

## **DERECHOS DE AUTORÍA**

Nosotros, René Javier Álvarez González y Sandra Isabel Shagñay Vimos, declaráramos ser los autores del presente trabajo de tesis **“SISTEMA RADAR PARA ANÁLISIS Y RECONSTRUCCIÓN DE PERFILES TOPOGRÁFICOS A MEDIA ESCALA”**, que fue elaborado en su totalidad por nosotros, bajo la dirección del Ingeniero Hugo Moreno Avilés, haciéndonos totalmente responsables por las ideas, criterios, doctrinas y resultados expuestos en esta Tesis, y el patrimonio de la misma pertenece a la Escuela Superior Politécnica de Chimborazo.

---

René Javier Álvarez González  
CI. 060409660-2

---

Sandra Isabel Shagñay Vimos  
CI. 060359415-1

## ÍNDICE DE ABREVIATURAS

|              |   |
|--------------|---|
| <b>ALS</b>   | Escáner Laser Aerotransportado  |
| <b>API</b>   | Application Programming Interface (Interfaz de Programación de Aplicaciones)                |
| <b>CIF</b>   | Common Intermediate Format (Formato Intermedio Común)                                       |
| <b>CMOS</b>  | Complementary metal-oxide-semiconductor<br>(Semiconductor complementario de óxido metálico) |
| <b>FFT</b>   | Fast Fourier Transform (Transformada Rápida de Fourier)                                     |
| <b>FPS</b>   | Frames per Second (Cuadros por segundo)   |
| <b>GPR</b>   | Radar de Penetración Terrestre  |
| <b>GUI</b>   | Interfaz Gráfica de Usuario   |
| <b>HF</b>    | High Frequency  |
| <b>IEEE</b>  | Instituto de Ingenieros Eléctricos y Electrónicos   |
| <b>LADAR</b> | Láser Detección And Ranging (Detección y medición de distancias por Laser)                  |
| <b>LIDAR</b> | Light Detection and Ranging (Detección y medición de distancias por Luz)                    |
| <b>MMW</b>   | Millimeter wave (Onda Milimetrica)  |
| <b>NIST</b>  | Instituto Nacional de Estándares y Tecnología   |
| <b>PSR</b>   | Primary Surveillance Radar (Radar de vigilancia primario)                                   |
| <b>QCIF</b>  | Quarter Common Intermediate Format (Medicion de Resolucion de Video)                        |
| <b>QVGA</b>  | Quarter Video Graphics Array()  |
| <b>RADAR</b> | Radio Detection And Ranging (Detección y medición de distancias por radio)                  |
| <b>RGB</b>   | Red Green Blue (Rojo Verde Azul)gg  |
| <b>RF</b>    | Radio Frecuencia  |
| <b>SSR</b>   | Secondary Surveillance Radar (Radar de vigilancia secundario)                               |
| <b>UHF</b>   | High Frequency (Frecuencia Ultra Alta)  |
| <b>USB</b>   | Universal Serial Bus  |
| <b>VHF</b>   | Very High Frequency (Muy alta frecuencia)   |



# ÍNDICE GENERAL

|                            |  |
|----------------------------|--|
| PORTADA                    |  |
| FIRMAS DE RESPONSABILIDAD  |  |
| RESPONSABILIDAD DE AUTORES |  |
| INDICE DE ABREVIATURAS     |  |
| INDICE GENERAL             |  |
| INDICE DE FIGURAS          |  |
| INDICE DE TABLAS           |  |
| INDICE DE ANEXOS           |  |
| INTRODUCCIÓN               |  |

## CAPÍTULO I

|                            |        |
|----------------------------|--------|
| 1. MARCO REFERENCIAL ..... | - 17 - |
| 1.1. ANTECEDENTES .....    | - 17 - |
| 1.1. JUSTIFICACIÓN .....   | - 19 - |
| 1.2. OBJETIVOS .....       | - 20 - |
| 1.2.1. GENERAL .....       | - 20 - |
| 1.2.2. ESPECIFICOS .....   | - 20 - |
| 1.3. HIPÓTESIS .....       | - 20 - |

## CAPÍTULO II

|  |        |
|--|--------|
| 2. MARCO TEÓRICO .....   | - 21 - |
| 2.1. RADAR .....   | - 21 - |
| 2.1.1. Definición .....  | - 21 - |
| 2.1.2. Principio de Funcionamiento de un Radar .....                             | - 22 - |
| 2.1.3. Clasificación Sistema Radar .....   | - 22 - |
| 2.1.3.1. Según el número de antenas: .....                                       | - 22 - |
| 2.1.3.2. Según el Blanco .....   | - 23 - |
| 2.1.3.3. Según su forma de Onda .....  | - 23 - |
| 2.1.3.4. Según su Finalidad .....  | - 24 - |
| 2.1.3. Frecuencias Radar .....   | - 24 - |
| 2.1.4. Radar Laser (LADAR) .....   | - 25 - |
| 2.1.4.1. Principios de Adquisición de Imágenes de Rango para Radares Laser ..... | - 26 - |
| 2.1.4.1.1. Principio de Fase .....   | - 27 - |
| 2.1.4.1.2. Correlación Cruzada .....   | - 27 - |
| 2.1.4.1.3. Triangulación Activa .....  | - 27 - |
| 2.1.4.1.4. Tiempo de vuelo .....   | - 27 - |
| 2.1.5. Análisis del Sistema Radar .....  | - 27 - |

|   |        |
|---|--------|
| 2.1.5.1. Resolución del Sistema .....                                   | - 28 - |
| 2.1.5.2. Distancia máxima de Análisis de Señal .....                    | - 28 - |
| 2.1.5.3. Señal de Onda Continua.....                                    | - 30 - |
| 2.1.5.3.1. Onda transmitida.....  | - 30 - |
| 2.1.5.3.2. Onda Recibida.....   | - 31 - |
| 2.1.5.3.3. Detección del Objeto .....                                   | - 32 - |
| 2.1.5.4. Señal Chirp.....   | - 34 - |
| 2.1.5.4.1. Señal transmitida.....                                       | - 34 - |
| 2.1.5.4.2. Señal Recibida .....   | - 35 - |
| 2.1.5.4.3. Detección del Objeto .....                                   | - 36 - |
| 2.2. PERFIL TOPOGRÁFICO .....   | - 38 - |
| 2.1.1. Definición .....   | - 38 - |
| 2.1.2. Tipos de Perfiles Topográficos .....                             | - 39 - |
| 2.2.1.1. Perfiles Longitudinales.....                                   | - 39 - |
| 2.2.1.2. Perfiles Transversales.....                                    | - 39 - |
| 2.1.3. Escalas .....  | - 39 - |
| 2.2.2. Técnicas Radar para Reconstrucción de Perfiles Topográficos..... | - 40 - |
| 2.3. SENSOR KINECT .....  | - 41 - |
| 2.1.4. Especificaciones Técnicas .....                                  | - 41 - |
| 2.1.5. Funcionamiento .....   | - 44 - |
| 2.3.4. Técnica de Profundidad .....                                     | - 45 - |

### **CAPÍTULO III**

|   |        |
|---|--------|
| 3. SISTEMA RADAR PARA RECONSTRUCCIÓN DE PERFILES TOPOGRAFICOS UTILIZANDO KINECT ..... | - 48 - |
| 3.1. SISTEMA RADAR UTILIZANDO KINECT .....  | - 48 - |
| 3.1.1. Señal Transmitida y Reflejada.....   | - 49 - |
| 3.1.2. Kinect.....  | - 49 - |
| 3.1.3. Open NI.....   | - 50 - |
| 3.1.3.1. Arquitectura.....  | - 51 - |
| 3.1.4. MATLAB.....  | - 52 - |
| 3.1.4.1. Características .....  | - 52 - |
| 3.1.4.2. Funcionamiento.....  | - 53 - |
| 3.1.4.3. Archivos MEX .....   | - 53 - |
| 3.1.4.3.1. Kinect-Mex .....   | - 54 - |
| 3.2. APLICACIÓN PARA RECONSTRUCCIÓN DE PERFILES TOPOGRÁFICOS.....                     | - 55 - |
| 3.2.1. Código en MATLAB .....   | - 55 - |
| 3.2.2. GUI en MATLAB.....   | - 60 - |

### **CAPÍTULO IV**

|                                 |        |
|---------------------------------|--------|
| 4. MEDICIONES Y RESULTADOS..... | - 65 - |
| 4.1. PRUEBAS DE DISTANCIA ..... | - 65 - |

|        |   |        |
|--------|---|--------|
| 4.1.1. | Distancia con Diferentes Materiales .....                     | - 65 - |
| 4.1.2. | Resolución y Frecuencia de Muestreo .....                     | - 66 - |
| 4.1.3. | Exactitud .....   | - 70 - |
| 4.1.4. | Compensación de Error (Método de los Mínimos Cuadrados) ..... | - 72 - |
| 4.2.   | PRUEBAS DE PERFIL .....                                       | - 75 - |
| 4.2.1. | Perfil a 1m.....  | - 76 - |
| 4.2.2. | Perfil a 2m.....  | - 77 - |
| 4.3.   | ANÁLISIS COMPARATIVO.....                                     | - 79 - |
| 4.3.1. | Perfil a 1m.....  | - 79 - |
| 4.3.2. | Perfil a 2m.....  | - 80 - |

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMARY

GLOSARIO

BIBLIOGRAFIA

ANEXOS

## INDICE DE FIGURAS

|  |        |
|--|--------|
| <b>FIGURA 1:</b> FUNCIONAMIENTO DE UN SISTEMA RADAR .....                      | - 22 - |
| <b>FIGURA 2:</b> FRECUENCIA Y TIPOS DE RADAR .....                             | - 24 - |
| <b>FIGURA 3:</b> SEÑAL TRANSMITIDA - REFLEJADA .....                           | - 29 - |
| <b>FIGURA 4:</b> SEÑAL TRANSMITIDA .....                                       | - 31 - |
| <b>FIGURA 5:</b> SEÑAL RECIBIDA .....  | - 32 - |
| <b>FIGURA 6:</b> MULTIPLICACIÓN SEÑAL TRANSMITIDA Y RECIBIDA .....             | - 33 - |
| <b>FIGURA 7:</b> TRANSFORMADA RÁPIDA DE FOURIER .....                          | - 33 - |
| <b>FIGURA 8:</b> SEÑAL CHIRP .....   | - 34 - |
| <b>FIGURA 9:</b> SEÑAL TRANSMITIDA (CHIRP) .....                               | - 35 - |
| <b>FIGURA 10:</b> SEÑAL RECIBIDA (CHIRP) .....                                 | - 36 - |
| <b>FIGURA 11:</b> MULTIPLICACIÓN SEÑAL TRANSMITIDA Y RECIBIDA (CHIRP)<br>..... | - 37 - |
| <b>FIGURA 12:</b> TRANSFORMADA RÁPIDA DE FOURIER .....                         | - 38 - |
| <b>FIGURA 13:</b> PERFIL TOPOGRÁFICO .....                                     | - 38 - |
| <b>FIGURA 14:</b> ESCALAS .....  | - 39 - |
| <b>FIGURA 15:</b> KINECT .....   | - 41 - |
| <b>FIGURA 16:</b> COMPONENTES KINECT .....                                     | - 42 - |
| <b>FIGURA 17:</b> KINECT LASER PRODUCT .....                                   | - 42 - |
| <b>FIGURA 18:</b> INTERIOR KINECT .....  | - 44 - |
| <b>FIGURA 19:</b> FUNCIONAMIENTO KINECT .....                                  | - 45 - |
| <b>FIGURA 20:</b> PROFUNDIDAD KINECT .....                                     | - 46 - |
| <b>FIGURA 21:</b> IMAGEN DE PROFUNDIDAD .....                                  | - 46 - |
| <b>FIGURA 22:</b> DIAGRAMA DE BLOQUES SISTEMA RADAR .....                      | - 49 - |
| <b>FIGURA 23:</b> ADAPTADOR KINECT .....                                       | - 50 - |
| <b>FIGURA 24:</b> ARQUITECTURA OPENNI .....                                    | - 51 - |
| <b>FIGURA 25:</b> INTERFAZ MATLAB .....  | - 52 - |
| <b>FIGURA 26:</b> DATOS KINECT .....   | - 56 - |
| <b>FIGURA 27:</b> SELECCIÓN DE DATOS KINECT .....                              | - 58 - |
| <b>FIGURA 28:</b> RECONSTRUCCIÓN DEL PERFIL .....                              | - 58 - |
| <b>FIGURA 29:</b> FLUJOGRAMA DEL PROGRAMA .....                                | - 59 - |
| <b>FIGURA 30:</b> GUI CREADA EN MATLAB .....                                   | - 60 - |
| <b>FIGURA 31:</b> NUEVO GUI GUIDE MATLAB .....                                 | - 61 - |
| <b>FIGURA 32:</b> VENTANA EDITOR GUIDE MATLAB .....                            | - 62 - |
| <b>FIGURA 33:</b> PROPIEDADES DE LOS COMPONENTES .....                         | - 62 - |
| <b>FIGURA 34:</b> GUI CON SUS COMPONENTES .....                                | - 63 - |
| <b>FIGURA 35:</b> CALLBACK .....   | - 63 - |
| <b>FIGURA 36:</b> SISTEMA RADAR GUI .....                                      | - 64 - |
| <b>FIGURA 37:</b> MEDICIÓN DE DISTANCIA .....                                  | - 65 - |
| <b>FIGURA 38:</b> ESCALA MEDIDA VS ESCALA REAL .....                           | - 71 - |

|   |        |
|---|--------|
| <b>FIGURA 39:</b> ERROR VS DISTANCIA .....                          | - 71 - |
| <b>FIGURA 40:</b> RECTA DE REGRESIÓN .....                          | - 72 - |
| <b>FIGURA 41:</b> MÉTODO DE LOS MÍNIMOS CUADRADOS.....              | - 73 - |
| <b>FIGURA 42:</b> FLUJOGRAMA (COMPENSACIÓN DE ERROR).....           | - 75 - |
| <b>FIGURA 43:</b> PRUEBA DE PERFIL .....                            | - 75 - |
| <b>FIGURA 44:</b> PERFIL 1M DESPLAZAMIENTO 0.30 M.....              | - 76 - |
| <b>FIGURA 45:</b> PERFIL 1M DESPLAZAMIENTO 0.10 M.....              | - 76 - |
| <b>FIGURA 46:</b> PERFIL 2M DESPLAZAMIENTO 0.30 M.....              | - 77 - |
| <b>FIGURA 47:</b> PERFIL 2M DESPLAZAMIENTO 0.10 M.....              | - 77 - |
| <b>FIGURA 48:</b> PERFIL DOS ELEMENTOS .....                        | - 78 - |
| <b>FIGURA 49:</b> PERFIL 2M DOS ELEMENTOS DESPLAZAMIENTO 0.10 M.... | - 78 - |
| <b>FIGURA 50:</b> DISTANCIA REAL-OBTENIDA 1M 0.30M .....            | - 79 - |
| <b>FIGURA 51:</b> DISTANCIA REAL-OBTENIDA 1M 0.10M .....            | - 79 - |
| <b>FIGURA 52:</b> DISTANCIA REAL-OBTENIDA 2M 0.30M .....            | - 80 - |
| <b>FIGURA 53:</b> DISTANCIA REAL-OBTENIDA 2M 0.10M .....            | - 80 - |
| <b>FIGURA 54:</b> DISTANCIA REAL-OBTENIDA DOS ELEMENTOS 2M 0.10M. - | - 81 - |

## INDICE DE TABLAS

|   |        |
|---|--------|
| <b>TABLA 1:</b> FRECUENCIAS Y TIPOS DE RADAR..... | - 25 - |
| <b>TABLA 2:</b> TÉRMINOS MEX.....                 | - 54 - |
| <b>TABLA 3:</b> ARCHIVOS MEX.....                 | - 55 - |
| <b>TABLA 4:</b> DATOS DE DISTANCIA.....           | - 57 - |
| <b>TABLA 5:</b> DISTANCIAS.....                   | - 66 - |
| <b>TABLA 6:</b> DISTANCIA 1M.....                 | - 66 - |
| <b>TABLA 7:</b> DISTANCIA 2M.....                 | - 67 - |
| <b>TABLA 8:</b> DISTANCIA 3M.....                 | - 67 - |
| <b>TABLA 9:</b> DISTANCIA 4M.....                 | - 68 - |
| <b>TABLA 10:</b> MEDIDAS COMPENSADAS.....         | - 74 - |

## INDICE DE ANEXOS

|   |         |
|---|---------|
| <b>ANEXO 1:</b> CÓDIGO RADAR SEÑAL NORMAL.....      | - 94 -  |
| <b>ANEXO 2:</b> CÓDIGO RADAR SEÑAL CHIRP .....      | - 95 -  |
| <b>ANEXO 3:</b> INSTALACIÓN OPEN-NI.....            | - 97 -  |
| <b>ANEXO 4:</b> CÓDIGO SISTEMA RADAR COMPLETO ..... | - 100 - |
| <b>ANEXO 5 :</b> CAPTURAS.....                      | - 106 - |

## INTRODUCCION

El presente proyecto está enmarcado en el desarrollo de un sistema radar para la reconstrucción de perfiles topográficos a media escala, permitiendo obtener un perfil similar al real.

Este trabajo describe el principio básico de un sistema radar, su funcionamiento, frecuencias a la que trabaja, tipos de radar, realizando un análisis del mismo. Se detalla el perfil topográfico, así mismo se toma en cuenta su clasificación y definición de sus escalas.

Para poder graficar dicho perfil nos ayudamos del sensor Kinect que inicialmente fue creado con el propósito de ser un periférico de videojuegos, sin embargo al obtener información de una distancia requerida para la reconstrucción del perfil, se aprovecho del sensor de profundidad laser que posee, donde se analizan los requerimientos necesarios para que este sea vinculado a un ordenador, así como el adaptador y las herramientas de software.

Seguidamente utilizando Matlab, OpenNI y ciertas librerías se crea un algoritmo que permite reunir los datos obtenidos para así poder reconstruir el perfil topográfico.

Por último se realizó las pruebas pertinentes donde se adquieren medidas que definen los parámetros con los que trabaja el sistema y se procede a realizar una comparación de los resultados obtenidos con los reales.

Este proyecto puede servir como punto de partida para muchos otros ya que al generar un software se puede utilizar en nuevas simulaciones o experimentos que añadan al programa nuevas o diferentes funcionalidades.



# **CAPÍTULO I**

## **1. MARCO REFERENCIAL**

### **1.1. ANTECEDENTES**

El estudio que llevó al desarrollo del Radar comenzó con las observaciones de Heinrich Rudolf Hertz, quién ya había demostrado en 1887 que las ondas electromagnéticas son capaces de ser reflejadas, de un modo parecido el comportamiento de los rayos luminosos. También comprobó que las ondas electromagnéticas se transmiten en el espacio libre a la velocidad de la luz.

Robert Watson-Watt fue quien hace el aporte definitivo para el descubrimiento y puesta en práctica del radar. Watson-Watt planteó la teoría que una onda de radio que choca con un avión en vuelo es reflejada y que este eco puede ser recibido en tierra, permitiendo poder determinar a qué distancia se encuentra el avión, su velocidad y su dirección.

El radar fue un factor decisivo en la segunda guerra mundial ayudando a Inglaterra a detectar los bombarderos alemanes enviados por Hitler.

Desde su aparición, el radar ha tenido un rápido desarrollo, constituyendo una de las ramas de la electrónica de mayor impacto por sus aplicaciones, tanto en la esfera civil como en la militar. Hoy en día el término radar es bastante popular y tiende aún, en muchos casos, a ser asociado al área militar. Si bien es cierto que su empleo con fines bélicos fue lo que impulsó el desarrollo del radar en sus inicios, pronto su uso se extendió a otros campos de la sociedad con fines pacíficos, como por ejemplo, la navegación aérea y marítima, la meteorología, el estudio de la Ionosfera, la investigación del espacio exterior, la geodesia y cartografía, etc.

En nuestro país existen proyectos como el nuevo sistema de radares de control para el tráfico aéreo que la Aviación Civil implementará. Se trata de un sistema integrado que proveerá el 95% de control aéreo desde todos los puntos del país, con un dispositivo en los aviones, permitiendo la visualización del espacio aéreo durante todo el desarrollo de la nave desde su despegue hasta su aterrizaje.

Los registros históricos más antiguos sobre Topografía que existen en nuestros días, afirman que esta ciencia se originó en Egipto. Herótodo manifestó que Sesostris dividió Egipto en lotes para el pago de impuestos. Las inundaciones anuales del río Nilo arrastraron partes de estos lotes y se designaron topógrafos para redefinir los linderos. Estos topógrafos antiguos se les llamaban estiracuerdas, debido a que sus medidas se hacían con cuerdas que tenían marcas unitarias a determinadas distancias. Las primeras cartas geográficas corresponden a Tales de Mileto y Anaximandro.

El desarrollo de la topografía en América se presenta en los tiempos de la conquista y la colonia, con los trabajos realizados por Mutis, Alexander Von Humboldt y Francisco José de Caldas. Los Estados Unidos tuvieron como presidente a un geómetra, George Washington, quien realizó la medición del territorio occidental de la colonia y de las llanuras de los montes Apalaches.

El Instituto Geográfico Militar del Ecuador es la institución técnica y científica, encargada de la elaboración de la Cartografía Nacional y del archivo de datos geográficos de nuestro país. Su actividad se enmarca en la Ley de la Cartografía Nacional y elabora mapas, cartas y demás documentos cartográficos oficiales del territorio ecuatoriano que son la base para la planificación de obras y trabajos vitales para el desarrollo y progreso de la Patria.

## **1.1. JUSTIFICACIÓN**

La Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo cuenta con el grupo de investigación en electromagnetismo y microondas que tiene uno de sus objetivos la investigación en sistemas radares y sus aplicaciones, por lo que se realizará una investigación en esta área.

Tradicionalmente, la información topográfica y geomorfológica ha venido haciéndose a partir de mapas topográficos y a través de la fotointerpretación de fotografías aéreas, dada la necesidad existente en el país por la obtención de perfiles topográficos a diferentes escalas de determinadas zonas para la construcción de obras, se investigó y desarrolló un sistema radar cuyos parámetros fueron analizados desde el punto vista matemático y físico, para luego utilizando el software MATLAB, se realizaron los cálculos y simulaciones necesarias para la implementación final.

La implementación del sistema radar se realizó usando el sensor Kinect que puede ser movido a través de una topografía compleja y permite una conexión fácil al computador gracias a la existencia de varios controladores, con este sensor se obtuvo información del espectro infrarrojo que fue analizada con el sistema desarrollado en MATLAB para la reconstrucción del perfil topográfico.

## **1.2. OBJETIVOS**

### **1.2.1. GENERAL**

Diseñar e Implementar un sistema radar para análisis y reconstrucción de perfiles topográficos a media escala.

### **1.2.2. ESPECIFICOS**

- Determinar los requerimientos de diseño para el sistema radar.
- Desarrollar algoritmos de análisis de señales de radar para la reconstrucción de perfiles topográficos.
- Simular el sistema radar bajo software MATLAB o LABVIEW.
- Implementar el sistema radar usando el sensor Kinect.
- Optimizar el sistema a través de pruebas de campo.

## **1.3. HIPÓTESIS**

Un sistema radar permitirá reconstruir perfiles topográficos a media escala?

## **CAPÍTULO II**

### **2. MARCO TEÓRICO**

#### **2.1. RADAR**

##### **2.1.1. Definición**

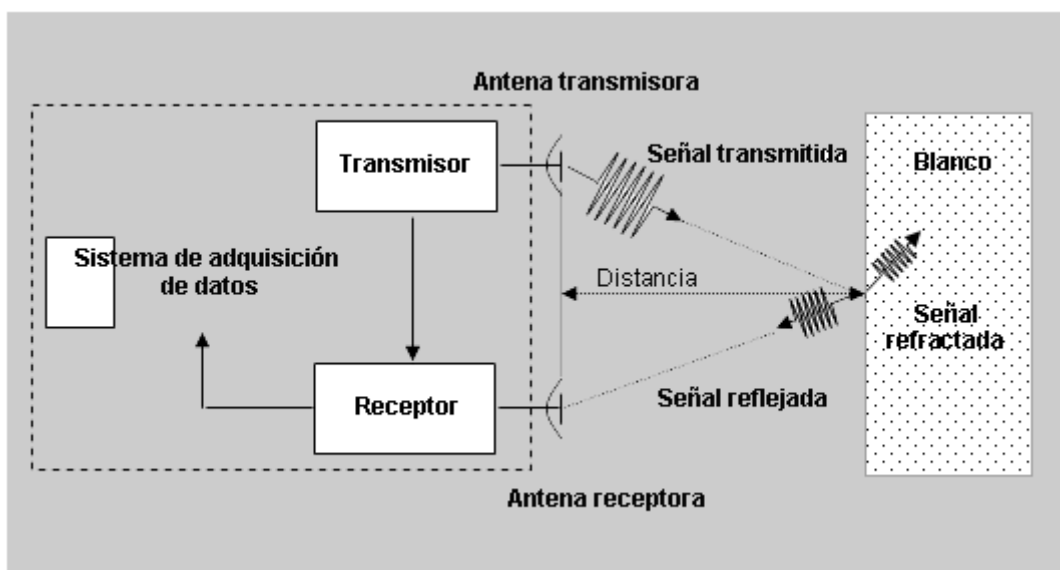
RADAR proviene del acrónimo inglés **Radio Detection And Ranging**, que es el proceso de transmitir, recibir, detectar y procesar una onda electromagnética que se refleja desde un objetivo y permite la detección inicial de su posición a través de la distancia de un objeto con respecto al sistema transmisor.

El uso de ondas electromagnéticas con diversas longitudes de onda detecta objetos más allá del rango de otro tipo de emisiones de la luz visible, sonido, etc. Su movimiento está basado en impulsos de radio, microondas que chocan con objetos y se reflejan en la misma posición del emisor. Partiendo del "eco" se puede tener gran cantidad de información.

## 2.1.2. Principio de Funcionamiento de un Radar

El principio básico es emitir una onda electromagnética o acústica y detectar el retorno reflejado desde el contorno de la superficie muestreada como se muestra en la figura 1. La señal de un radar es generada en el transmisor y radiada en dirección de un objeto donde es reflejada hacia el receptor, el mismo que procesa y obtiene la información que puede ser la distancia, altura, velocidad, dirección y características del objeto.

**FIGURA 1: FUNCIONAMIENTO DE UN SISTEMA RADAR**



Fuente: <http://www.umag.cl/investigacion/dpa/radar/sistemas/ecuaciones/figura-02.gif>

El tipo de radar depende de la función que requiera, para especificar un radar en común se debe conocer el modo de operación, el método de transmisión y que va a realizar.

## 2.1.3. Clasificación Sistema Radar

### 2.1.3.1. Según el número de antenas:

- Monoestático: consta de una sola antena que hace de transmisor y receptor al mismo tiempo y su ubicación es estática.

- Biestático: Este sistema tiene dos antenas, una para la transmisión y otra para la recepción, los mismos que pueden actuar en un mismo o diferente tiempo. Su posición es estática
- Multiestático: la información recibida proviene de varias antenas que se encuentran estaticas

### **2.1.3.2. Según el Blanco**

- Radar Primario PSR (Primary Surveillance Radar): Este radar transmite ondas electromagnéticas con tiempos de duración cortos, las mismas que se reflejan sobre los objetos en los que choca, siendo captadas nuevamente por el sistema PSR, obteniendo una señal de retorno. La aplicación típica es el sistema radar para control del tráfico aéreo.
- Radar Secundario de Vigilancia SSR (Secondary Surveillance Radar): Este sistema envía una señal codificada solicitando identificación del objeto. El objeto responde al poseer un transponder que permite el procesamiento y transmisión de las señales codificadas.

Es importante tener conocimiento de este tipo de radares, ya que el radar primario requiere de un transmisor y un receptor para detectar el objeto, siendo utilizados para vigilancia y búsqueda. Los radares secundarios además de un trasmisor y receptor necesitan un transponder donde este equipo procese y transmita la señal, permitiendo emplearlos en el control de tráfico aéreo comercial.

### **2.1.3.3. Según su forma de Onda**

- Radar de onda continua: emite la señal interrumpidamente y utiliza la teoría del Efecto Doppler para determinar la posición y velocidad del objeto en análisis
- Radar de onda pulsada: transmite periódicamente un pulso, que puede ser modulado o no.

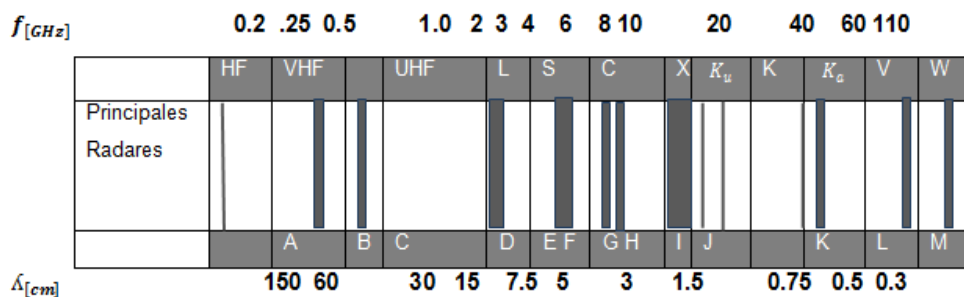
#### 2.1.3.4. Según su Finalidad

- Radar de Seguimiento: tiene la capacidad de seguir el movimiento de un blanco.
- Radar de Búsqueda: muestra los blancos explorando todo el espacio.

#### 2.1.3. Frecuencias Radar

Un sistema radar trabaja en diferentes frecuencias, dependiendo de la aplicación. La IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) ha asignado bandas con letras como se muestra en la Figura 2.

**FIGURA 2: FRECUENCIA Y TIPOS DE RADAR**



**Fuente:** BASSEM R., Radar Systems Analysis and Design Using MATLAB, 3ª ed., Alabama-USA., CRC Press., 2013., Pp. 4.

Como se observa en la figura 2, las frecuencias para las bandas VHF, UHF y L, se utiliza para radares de exploración y búsqueda. Seguidamente en las frecuencias L, S, C, se aplica para búsqueda y seguimiento de objetos. En la banda C y X permite controlar el fuego y obtener imágenes. Finalmente en las frecuencias X, Ku, K, Ka y W, trabajan para la detección de misiles. La tabla 1 describe los rangos de frecuencias asignadas por la IEEE.



**TABLA 1: FRECUENCIAS Y TIPOS DE RADAR**

| Designación de Frecuencias | Rango de frecuencias en GHz(estándar IEEE) | Rango de frecuencias en GHz(estándar IEEE) |
|----------------------------|--|--|
| <i>HF</i>                  | <i>0.003 – 0.03</i>                        | <i>A</i>                                   |
| <i>VHF</i>                 | <i>0.03 - 0.3</i>                          | <i>A&lt;0.25; B&gt;0.25</i>                |
| <i>UHF</i>                 | <i>0.3 – 1.0</i>                           | <i>B&lt;0.5; c&gt;0.5</i>                  |
| <i>L_Band</i>              | <i>1.0 – 2.0</i>                           | <i>D</i>                                   |
| <i>S_Band</i>              | <i>2.0 – 4.0</i>                           | <i>E&lt;3.0; F&gt;3.0</i>                  |
| <i>C_Band</i>              | <i>4.0 – 8.0</i>                           | <i>G&lt;6.0; H&gt;6.0</i>                  |
| <i>X_Band</i>              | <i>8.0 – 12.5</i>                          | <i>I&lt;10.0; J&gt;10.0</i>                |
| <i>Ku_Band</i>             | <i>12.5 – 18.0</i>                         | <i>J</i>                                   |
| <i>K_Band</i>              | <i>18.0 - 26.5</i>                         | <i>J&lt;20.0; K&gt;20.0</i>                |
| <i>Ka_Band</i>             | <i>26.5 – 40.0</i>                         | <i>K</i>                                   |
| <i>V &amp; W o MMW</i>     | <i>Normal &gt;34.0</i>                     |  |

**Fuente:** BASSEM R., Radar Systems Analysis and Design Using MATLAB, 3ª ed., Alabama-USA., CRC Press., 2013., Pp. 4.

#### **2.1.4. Radar Laser (LADAR)**

Las aplicaciones y las técnicas radar se expandieron considerando varios aspectos . La longitud de onda de la señal transmitida, es una de ellas que se considero para su desarrollo, técnicamente se utiliza desde los 50 cm hasta descender hacia regiones milimétricas y microondas.

Fuentes de luz pulsada y detectores ópticos se utilizaron por primera vez en 1938 para medir la altura base de las nubes. El acrónimo LIDAR ( Light Detection and Ranging) se utilizó por primera vez en 1953, y en 1962 el desarrollo de alta energía o conmutación de láseres Q pulsados hicieron tales fuentes disponibles para aplicaciones LIDAR. En 1963, Fiocco y Sullivan publicaron trabajos sobre observaciones atmosféricas utilizando un rubí laser.

El NIST (Instituto Nacional de Estándares y Tecnología) acogió el término LADAR (Láser Detección And Ranging) para éstos tipos de radar basados en sistemas Laser.

El sistema LiDAR está compuesto por un sensor eficaz, el mismo que envía un haz de laser hacia el objeto que desea medir, en el cual se refleja dicho rayo, siendo percibido por el sensor, que lo adopta y lo graba. Para obtener la distancia entre el sensor y el objeto, se mide el tiempo de la transmisión y recepción del pulso. Al multiplicar el tiempo y la velocidad de la luz láser produce un valor de ida y vuelta que es dos veces la distancia. El sensor identifica la dirección del rayo para calcular las coordenadas del objeto.

Todos los sistemas de detección de rango sean RADAR, LiDAR, o LADAR, funcionan con la transmisión y recepción de energía electromagnética. Su diferencia entre estos sistemas es que trabajan en diferentes frecuencias. Tomando en cuenta las mismas consideraciones, como el tiempo de propagación y la teoría de la antena, aplicados en estos sistemas.

El LADAR en si se limita a la longitud de onda que suelen ser creados mediante la transmisión óptica del láser.

Los sistemas ladares han pasado a ser instrumentos de gran importancia en la investigación donde el alcance, el área de cobertura, la portabilidad ha ocupado el interés de científicos en mantener el láser y los elementos ópticos con un pico de eficiencia en el desarrollo de este sistema.

El sistema Ladar se aplica en arqueología, geografía, teledetección, geomorfología, arqueología y cartografía.

#### **2.1.4.1. Principios de Adquisición de Imágenes de Rango para Radares Laser**

Los radares láser de adquisición de imágenes pueden usar diferentes principios:

#### **2.1.4.1.1. Principio de Fase**

Este principio se base en la emisión de un haz láser de amplitud modulada y la medición de la diferencia de fase entre las señales transmitidas y las recibidas.

#### **2.1.4.1.2. Correlación Cruzada**

El sistema radar de correlación cruzada se basa en el procesamiento de señales. La correlación cruzada es una forma de medir la similitud entre dos señales, por lo general usada para encontrar ciertas características relevantes en una señal que se desconoce por medio de la comparación con otra que sí se tiene conocimiento.

#### **2.1.4.1.3. Triangulación Activa**

El método de triangulación activa se denomina así porque el punto donde ilumina el láser, el emisor laser y la cámara forman un triángulo, que permite determinar la profundidad de un objeto a través de un sensor activo que lo ilumina con un láser proyectado desde un punto y captado por un sensor desde el otro punto, tiene mayor precisión y trabaja en el orden de los metros.

#### **2.1.4.1.4. Tiempo de vuelo**

Los sistemas que están basados en el principio de tiempo de vuelo, utiliza un láser para medir distancias del transmisor a cada punto del objeto. Al tomar el tiempo en que se tarda el pulso de luz en viajar desde el dispositivo hasta el objeto, se puede medir la distancia y conocer la precisión del escáner dado a que se conoce la velocidad de la luz. El tiempo de vuelo tiene una precisión de orden milimétrico y opera hasta kilómetros.

#### **2.1.5. Análisis del Sistema Radar.**

Básicamente se inicia el proceso desde un análisis matemático para el desarrollo del sistema radar, donde se describe las características fundamentales de la señal.

### 2.1.5.1. Resolución del Sistema

La resolución de un sistema radar es el valor de la mínima variación de detección de la posición de un objeto. Para ello es necesario considerar la relación existente entre el ancho de banda, la frecuencia de muestreo y la frecuencia máxima dada por:

$$fs = 2 * fmax = B \quad (2.1)$$

Donde la exactitud de la distancia:

$$e_d = c * ts \quad (2.2)$$

$$ts = \frac{1}{fs} \quad (2.3)$$

$$e_d = \frac{c}{2 * fmax} \quad (2.4)$$

Obteniendo como resultado:

$$\Delta R = e_d = \frac{c}{B} = \frac{c}{2 * fmax} \quad (2.5)$$

Que es el valor de la resolución del sistema.

### 2.1.5.2. Distancia máxima de Análisis de Señal

La distancia máxima teórica a la que puede ser analizada la señal recibida en el sistema radar es:

$$dmaxteorica = \frac{c * t_f}{2} \quad (2.6)$$

Siendo  $t_f$  el tiempo final.

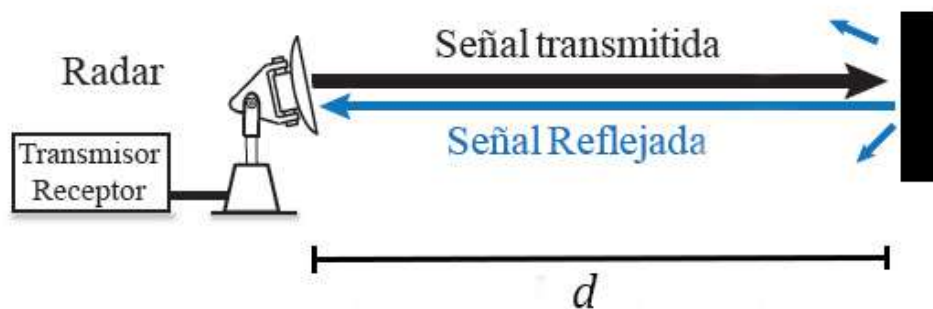
Para objetivos prácticos y para un mejor análisis de la señal tenemos que:

$$2 * d_{max} = \frac{c * t_f}{2} \quad (2.7)$$

$$d_{max} = \frac{c * t_f}{4} \quad (2.8)$$

En la simulación se relaciona la distancia desplazada de la señal recibida con las siguientes ecuaciones como se detalla en la figura 3.

**FIGURA 3: SEÑAL TRANSMITIDA - REFLEJADA**



Fuente: Álvarez R., Shagñay S.

$$desp = 2 * d \quad (2.9)$$

$$tdesp = \frac{desp}{c} \quad (2.10)$$

$$N = \frac{tdesp}{ts} \quad (2.11)$$

### 2.1.5.3. Señal de Onda Continua

#### 2.1.5.3.1. Onda transmitida

Para generar la señal se partió de modelar la señal, por medio de las ecuaciones que permitieron procesar una señal sinusoidal.

En este caso se utilizó la función coseno, tratándose de una señal análoga, donde sus valores oscilan en una rama de opciones prácticamente infinita.

Se graficó la onda con la ayuda de MATLAB en función de un vector de tiempo  $t$  y una frecuencia de muestreo:

$$f_s = 2 * f_{max} \quad (2.12)$$

.

Tomando como velocidad de propagación a la velocidad de la luz.

$$V_p = c = 3e8 \text{ m/s} \quad (2.13)$$

Siendo la señal transmitida:

$$V_t(t) = V_0 \cos(2\pi f t + \varphi_0) \quad (2.14)$$

Donde:

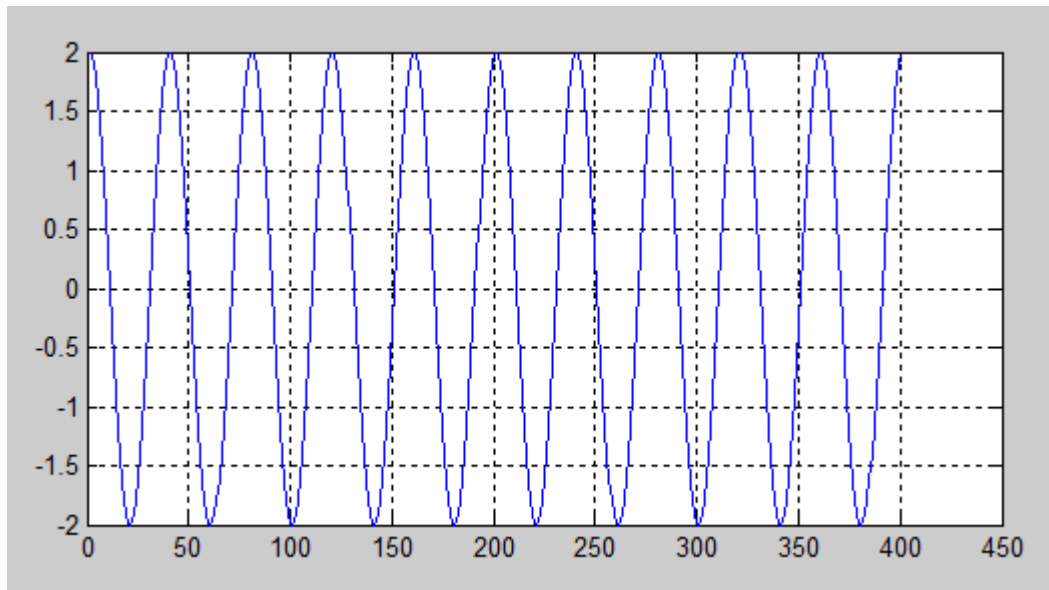
$f = \text{frecuencia}$

$t = \text{tiempo}$

$\varphi_0 = \text{Angulo de desfase}$

Graficando la onda transmitida a través de las ecuaciones programadas por código MATLAB disponible en el anexo 1, considerando el ángulo de desfase  $\varphi_0$  igual a cero y la amplitud  $V_0$  igual a 2 se obtiene la señal que se muestra en la Figura 4.

**FIGURA 4: SEÑAL TRANSMITIDA**



**Fuente:** Álvarez R., Shagñay S.

Como se observa en la figura 4 que la onda describe una curva continua. De hecho, esta onda es la gráfica de la función matemática coseno que oscilando dentro de un rango a una cierta frecuencia. Como las ondas electromagnéticas.

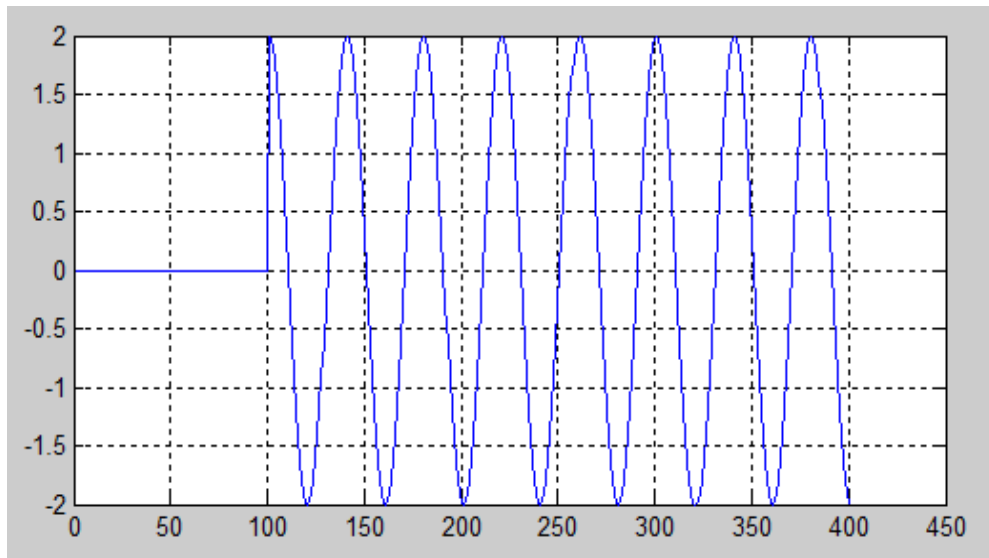
### 2.1.5.3.2. Onda Recibida

La onda que se transmite, se propaga en el espacio libre, choca contra un objeto, el mismo que refleja la onda y genera un desfase y un retardo de tiempo, cuyos valores se describen en la ecuación 2.15.

$$V_r(t) = V_0 K \cos(2\pi f(t - 2t_r) + \varphi_0) \mu(t - 2t_r) \quad (2.15)$$

Donde  $t_r$  es el retardo de la señal producto del tiempo de ida y regreso de la señal y  $\varphi_0$  es el desfase generado en la señal reflejada.

**FIGURA 5: SEÑAL RECIBIDA**



*Fuente:* Álvarez R., Shagñay S.

Para tener la señal recibida se retrasa la señal transmitida un determinado tiempo.

### 2.1.5.3.3. Detección del Objetivo

La teoría del radar indica que para la detección de la posición de un objeto es necesaria la multiplicación de las señales transmitida y recibida considerando un tiempo de origen común para las dos y cuya solución es la multiplicación de las señales correspondientes a las ecuaciones 2.14 y 2.15 se genera la señal final:

$$V_f(t) = V_t \cdot V_r \quad (2.16)$$

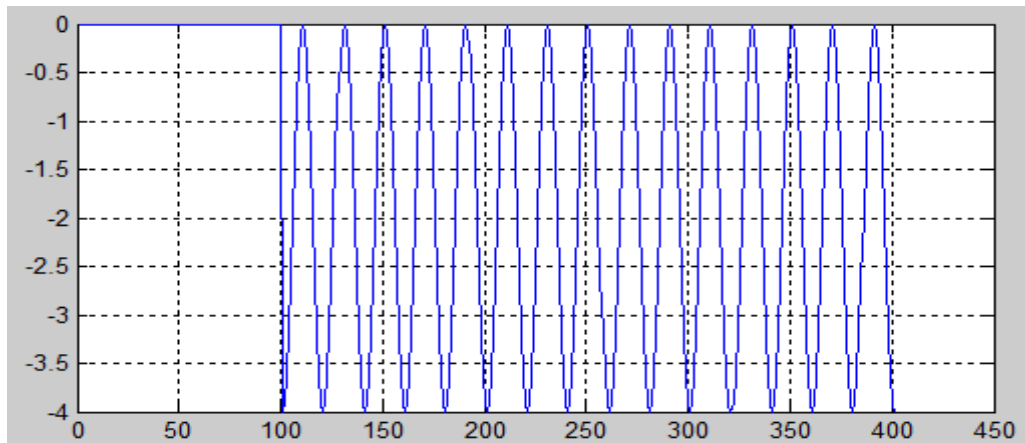
Obteniendo la ecuación 2.17.

$$V_f(t) = V_0 \cos(2\pi f t + \varphi_0) \cdot V_0 \cos(2\pi f(t - 2t_r) + \varphi_0) \mu(t - 2t_r) \quad (2.17)$$

Esta señal puede ser analizada desde el punto de vista trigonométrico para determinar las frecuencias existentes en la misma. La figura 6 muestra la señal recibida.



**FIGURA 6: MULTIPLICACIÓN SEÑAL TRANSMITIDA Y RECIBIDA**

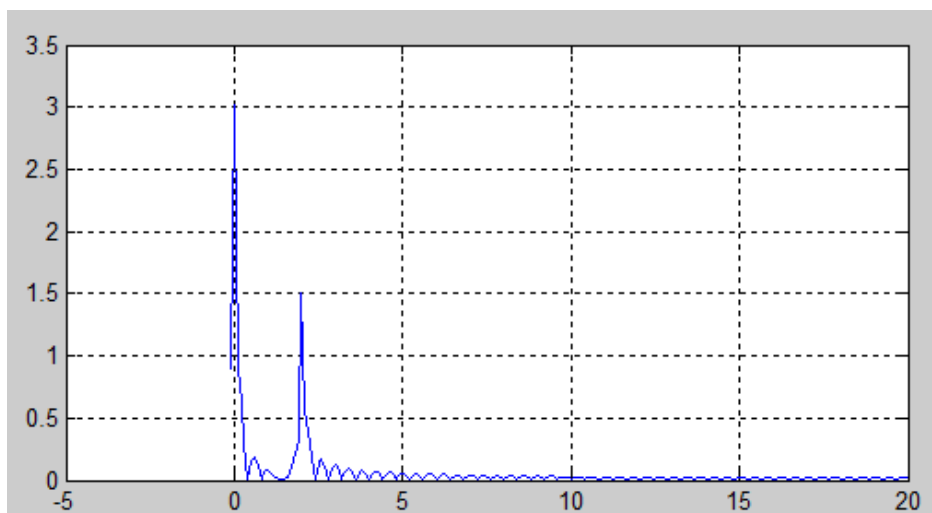


Fuente: Álvarez R., Shagñay S.

A la multiplicación de la señal transmitida con la recibida se le aplica la FFT (Transformada Rápida de Fourier) que es un método matemático para transformar la función del tiempo a función de la frecuencia. Es muy útil para el análisis de los fenómenos dependientes del tiempo.

$$fft(V_f)$$

**FIGURA 7: TRANSFORMADA RÁPIDA DE FOURIER**



Fuente: Álvarez R., Shagñay S.

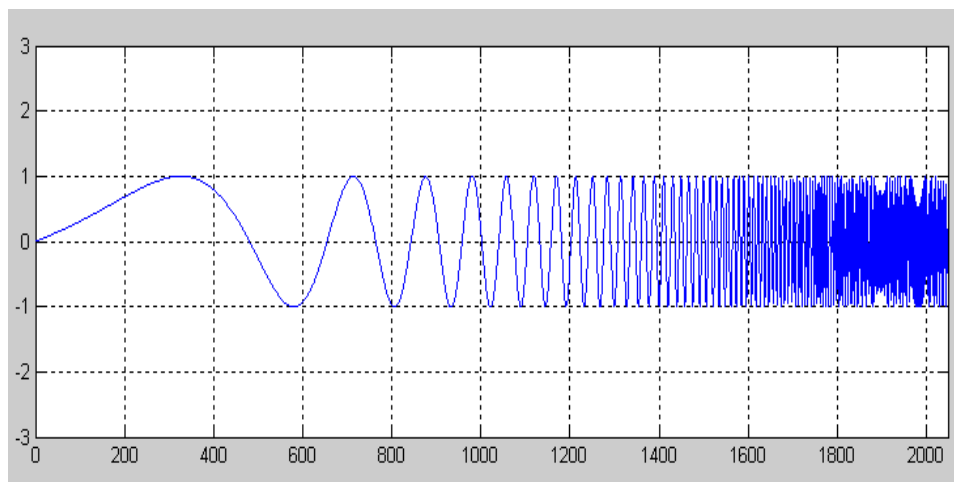
Luego del análisis de las ecuaciones mencionadas anteriormente en la figura 7 se detalla los picos obtenidos de la transformación del dominio del tiempo al dominio de la

frecuencia donde se observó un pico que representa al objeto detectado, más adelante en el análisis de la señal chirp se explica la relación del pico con la distancia a la que se encuentra el objeto.

#### 2.1.5.4. Señal Chirp

Es la señal que cambia su frecuencia en un periodo determinado de tiempo. La frecuencia de la señal chirp puede variar desde una frecuencia baja a una alta (up-chirp) o desde una frecuencia alta a una baja (low-chirp). Para su análisis también se dispone de código MATLAB disponible en el anexo 2. La figura 8 muestra una señal up-chirp.

**FIGURA 8: SEÑAL CHIRP**



*Fuente: Álvarez R., Shagñay S.*

##### 2.1.5.4.1. Señal transmitida

La ecuación para generar una señal chirp con amplitud  $A$ , frecuencia angular  $\omega_0$  y fase constante  $\phi_0$  esta dada por la ecuación 2.18.

$$V(t) = V_0 \cos(2\pi f(t)t + \phi_0) \quad (2.18)$$

Siendo  $f(t)$  la función de la frecuencia variante en el tiempo y que viene dada por:

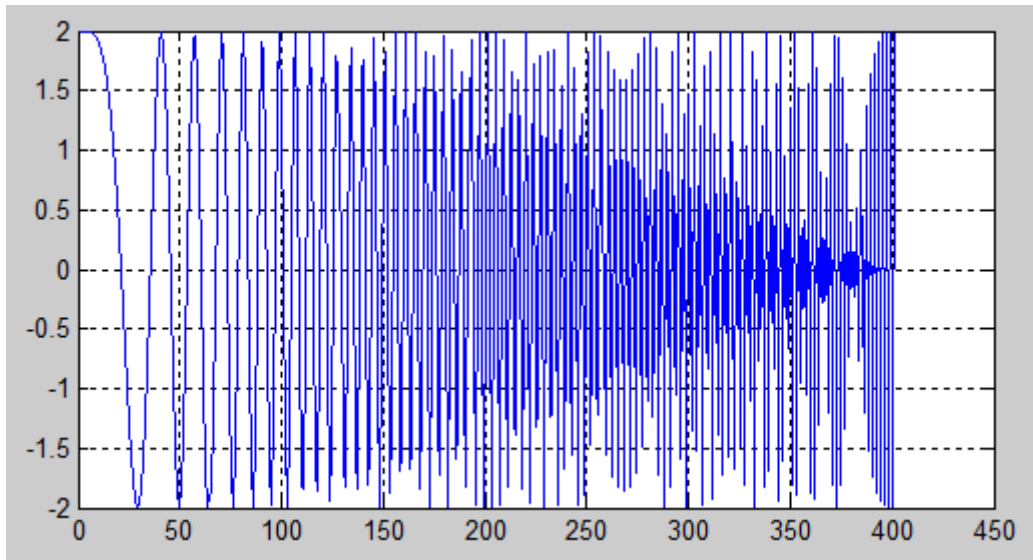
$$f(t) = \frac{k}{2}t + f_0 \quad (2.19)$$

Donde,  $k$  que es la rapidez con que cambia la frecuencia y se expresa como:

$$k = \frac{(f_1 - f_0)}{T} \quad (2.20)$$

En donde,  $f_0$  es la frecuencia inicial de la onda,  $f_1$  es la frecuencia al final de la duración  $T$ .

**FIGURA 9: SEÑAL TRANSMITIDA (CHIRP)**



**Fuente:** Álvarez R., Shagniay S.

La figura 9 describe la señal transmitida donde se observa el aumento progresivo de la frecuencia.

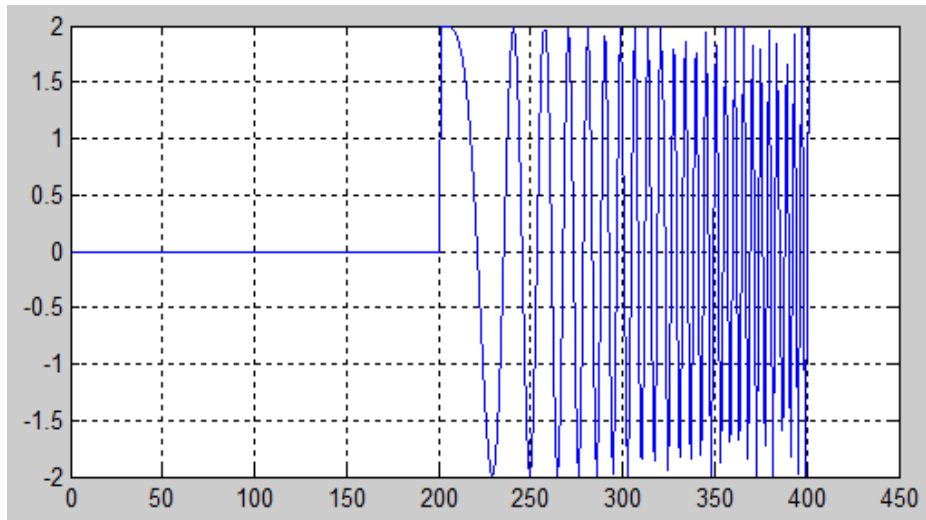
#### 2.1.5.4.2. Señal Recibida

La señal recibida es una señal similar a la transmitida, pero retardada en el tiempo. La ecuación 2.21 muestra su descripción considerando que no existen pérdidas en el espacio.

$$V_r(t) = V_0 K \cos \left( 2\pi \left( \frac{k}{2}(t - 2t_r) + f_0 \right) (t - 2t_r) + \varphi_0 \right) \mu(t - 2t_r) \quad (2.21)$$

La figura 10 muestra la señal recibida, la misma que fue simulada en MATLAB.

**FIGURA 10: SEÑAL RECIBIDA (CHIRP)**



Fuente: Álvarez R., Shagñay S.

Así mismo para tener la señal recibida se retrasa la señal transmitida un determinado tiempo.

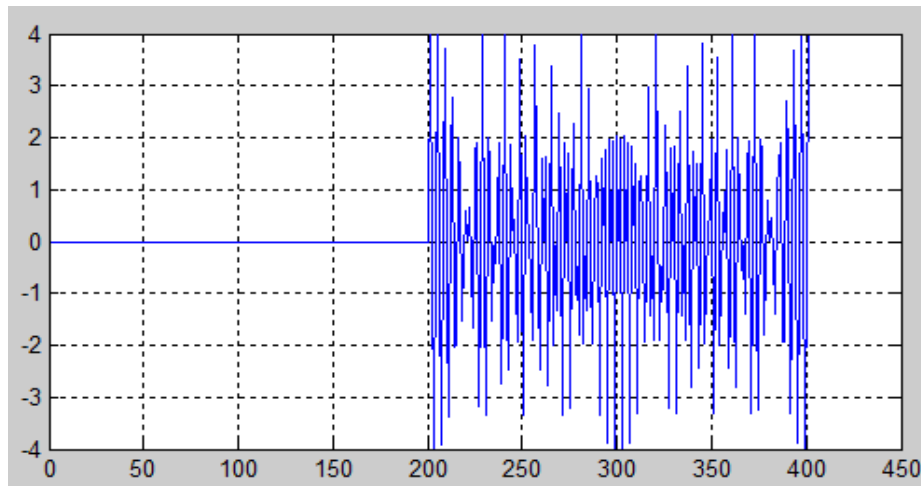
#### 2.1.5.4.3. Detección del Objeto

En base a la teoría expuesta, es necesario multiplicar la señal transmitida con la señal recibida obteniendo:

$$V_f(t) = V_0 \cos(2\pi f(t)t + \phi_0) \cdot V_0 K \cos\left(2\pi \left(\frac{k}{2}(t - 2t_r) + f_0\right) (t - 2t_r) + \phi_0\right) \mu(t - 2t_r) \quad (2.22)$$

La figura 11 detalla en forma gráfica la multiplicación de la señal transmitida y recibida.

**FIGURA 11: MULTIPLICACIÓN SEÑAL TRANSMITIDA Y RECIBIDA (CHIRP)**



Fuente: Álvarez R., Shagñay S.

Aplicando la transformada rápida de Fourier FFT se obtiene el espectro de amplitud, donde los picos del espectro están relacionados con la posición del objetivo.

Se determina la distancia del objetivo relacionado la frecuencia muestreada con la distancia a través de las siguientes ecuaciones:

$$\frac{fs}{2} = \frac{N}{2} * \Delta f \quad (2.23)$$

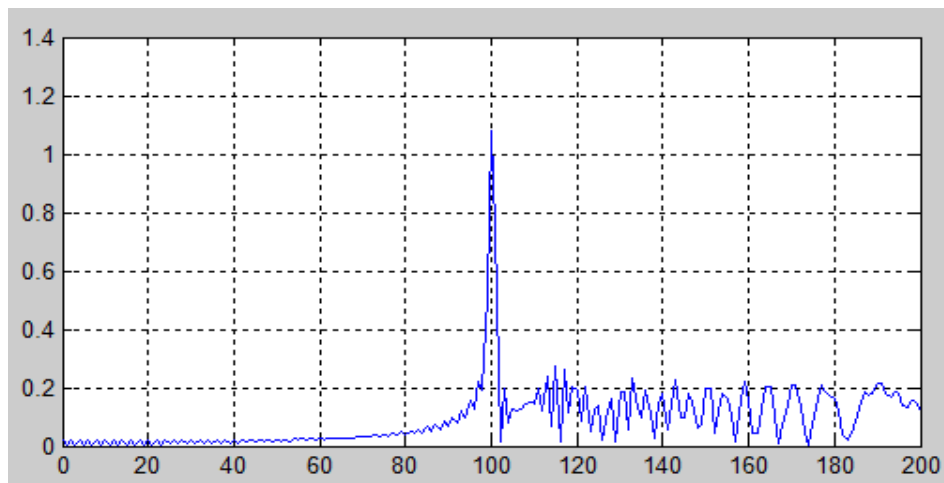
$$\Delta f = \frac{fs}{N} \quad (2.24)$$

$$\Delta r = \frac{c}{fs} \quad (2.25)$$

$$\Delta r = \frac{c}{N * \Delta f} \quad (2.26)$$

$$fft(V_f)$$

**FIGURA 12:** TRANSFORMADA RÁPIDA DE FOURIER



Fuente: Álvarez R., Shagñay S.

Por tanto en la figura 12 muestra la distancia a la que se encuentra el objeto, siendo representada por el pico más alto.

## 2.2. PERFIL TOPOGRÁFICO

### 2.1.1. Definición

El perfil Topográfico es la representación gráfica lineal del relieve del terreno, resultado de cortarlo por un plano vertical u horizontal. La figura 13 describe un perfil topográfico cortado por un plano vertical

**FIGURA 13:** PERFIL TOPOGRÁFICO



Fuente:<http://www.juntadeandalucia.es/medioambiente/servtc5/ventana/mostrarFicha.do?jsessionid=2EADB7B760403FDAA71FD F6A1D678516?idEquipamiento=28393>

## 2.1.2. Tipos de Perfiles Topográficos

### 2.2.1.1. Perfiles Longitudinales

Los perfiles longitudinales son el resultado de la intersección de la superficie del terreno con un plano vertical u horizontal. Para su representación se utilizan los ejes cartesianos X y Y. El eje X indica la distancia desde el origen y el eje Y la altitud. La escala de los ejes pueden ser diferentes, por lo general se tiende a aumentar los valores en el eje Y, logrando llegar hasta una diez veces la del eje X.

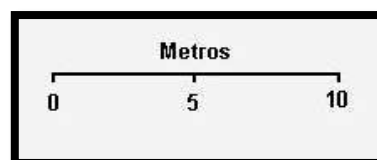
### 2.2.1.2. Perfiles Transversales

Los perfiles transversales son cortes verticales del terreno realizados perpendicularmente al perfil longitudinal. Se representa en los ejes cartesianos X y Y. En el eje de las X se representan las distancias con valores positivos y negativos. Mientras que en el eje de las Y se marcan las altitudes. Con respecto a la escala no suelen existir diferencias entre los ejes.

### 2.1.3. Escalas

La escala es la relación entre la distancia que existe en la representación gráfica y la distancia real correspondiente al terreno (ver Figura 14).

**FIGURA 14:** ESCALAS



**Fuente:** Álvarez R., Shagñay S.

Considerándose también como media escala la distancia en metros y gran escala la distancia en kilómetros.

### **2.2.2. Técnicas Radar para Reconstrucción de Perfiles Topográficos**

Una tecnología para perfiles topográficos es a través de un láser infrarrojo que hace el escaneo de objetos con el fin de registrar datos precisos para un posterior perfil.

Este laser está representado por un cuadro de puntos que captura la forma del objeto, en el cual se precisa una técnica que permita la reconstrucción de los perfiles.

Para mejor comprensión se detalla las diferentes técnicas radar para reconstrucción de perfiles topográficos.

- **Escáner Laser Aerotransportados (ALS)**

Permite obtener Modelos Digitales de Terreno por lo que se basa en un sensor laser que sobrepuesto en plataformas móviles como un avión o helicóptero hace un barrido de una superficie utilizando procedimientos fotográficos.

- **Radar de Penetración Terrestre (GPR)**

Esta técnica hace uso de un georadar que permite hacer el sondeo de profundidades hasta los 300 metros, que posee antenas de 10 a 15 MHz donde revela fallas geológicas tectónicas del terreno.

- **GeoSAR**

Es un sistema que mapea superficies de terrenos a nivel mundial, utiliza una tecnología radar de apertura sintética que proporciona imágenes de alta resolución y datos de terrenos de áreas extremadamente inaccesibles generando mapas topográficos



### 2.3. SENSOR KINECT

Es un sistema creado por la empresa Microsoft que permite a los usuarios interactuar con la consola de videojuegos Xbox 360 sin tener contacto físico con un controlador. Fue lanzado en noviembre del 2010.

Utiliza tecnología de una cámara RGB (ver Figura 15) que reconoce gestos, objetos e imágenes y comandos de voz.

**FIGURA 15:** KINECT



**Fuente:** <http://otakucity.net/wp-content/uploads/2011/06/kinect-xbox-360.jpg>

Este dispositivo es capaz de obtener la profundidad de los objetos. La aceptable precisión y su bajo costo permiten ser utilizado en diferentes aplicaciones.

#### 2.1.4. Especificaciones Técnicas

El Kinect es de forma rectangular de 30 cm, plástico negro donde la cámara y el láser se conectan a una placa metálica que paralelamente se conecta a dos microchips encargados de procesar y controlar la información obtenida por los sensores, destacando el de profundidad.

**FIGURA 16: COMPONENTES KINECT**



**Fuente:** <http://canaltech.com.br/o-que-e/kinect/Como-funciona-o-Kinect/>

Dispone de un micrófono Multi-array compuesto por la unión de cuatro micrófonos montados en un micrófono capaz de reconocer el audio, localizando la procedencia del mismo, apto para captar y eliminar el ruido del ambiente, (ver Figura 16).

Proyector Láser con las siguientes características:

- Láser de clase uno (ver Figura 17), producto seguro en la mayoría de condiciones de utilización, incluyendo el uso de ciertos instrumentos ópticos en visión directa.
- Su longitud de onda de 830 nm.
- Sistema de difracción subdividido en múltiples peticiones, para proyectar un patrón de puntos pseudo aleatorios.
- El láser posee una potencia de 60mW.

**FIGURA 17: KINECT LASER PRODUCT**



**Fuente:** Álvarez R., Shagñay S.

Sensor Infrarrojo que determina un CMOS monocromo con las siguientes propiedades:

- Tamaño de pixel 5.2 um
- Video de 5:4 con una resolución 1280x1204 Mp.
- 30 fps programables
- Temperatura en un rango de 0° a 70°C
- En la misma frecuencia del láser dispone de un filtro de paso infrarrojo. Una mínima influencia en el sensor si choca con luz visible de hasta 950 nm
- El sensor infrarrojo que está dentro del rango del espectro electromagnético de infrarrojo cercano entre 780 nm a 2500 nm por la longitud de onda.
- Para visión horizontal la cámara es de 57° y la distancia máxima de la cámara de profundidad estimado de 0.8 metros a 4 metros, un rango óptimo de 1,2 metros a 3,5 metros.

Tiene una cámara RGB, con un sensor de imagen CMOS con las siguientes especificaciones técnicas:

- Tamaño de pixel 2,8 unx2,8 um
- 1,3 Megapíxeles
- Resolución 1.280Hx1.024V
- Filtro de color Bayer
- 15 fps a máxima resolución, 30 fps a 640 x 512
- Rango de temperatura de -30° C a 70°C
- Soporta VGA, QVGA, CIF y QCIF.

Posee un motor de inclinación de  $\pm 27^\circ$  con un acelerómetro que indica la inclinación cada instante. Los componentes se encuentran dispuestos como se muestra en la Figura 18.

**FIGURA 18: INTERIOR KINECT**



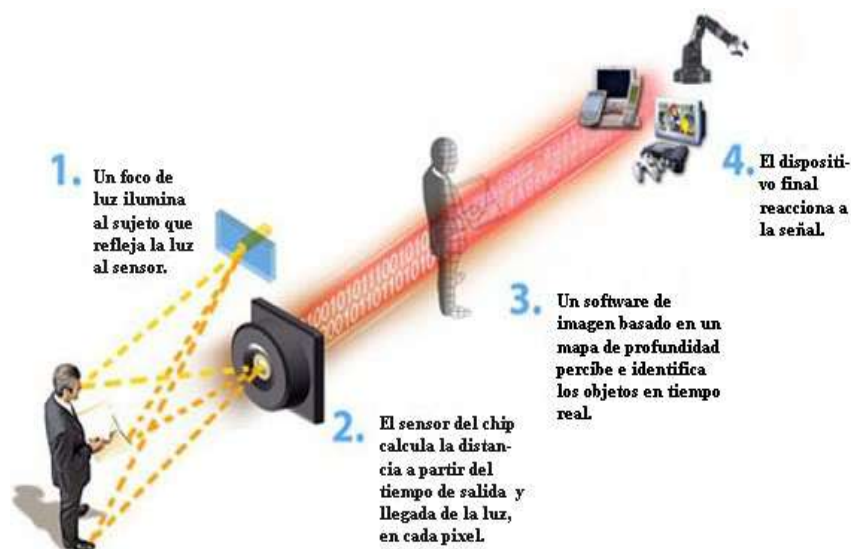
**Fuente:** <http://www.1080b.com/videojuegos-hd/microsoft-obtiene-grandes-ganancias-con-kinect/9358/>

### **2.1.5. Funcionamiento**

De las características mencionadas anteriormente se hace énfasis en el sensor de profundidad, siendo la principal clave de funcionamiento tomada por expertos mas la poca información dada por los creadores de Kinect, no se puede afirmar el método empleado para la captura de imágenes. Ya que el tipo concreto de algoritmo usado por Kinect es una información que PrimeSense, el fabricante del circuito integrado responsable de analizar la información del sensor de profundidad, no ha desvelado actualmente.

Sin embargo se sabe que dispone de un transmisor laser y un receptor que capta las señales reflejadas del láser (ver Figura 19). Podría tratarse de un sensor de Tiempo de Vuelo, donde la distancia se obtiene del tiempo de ida y vuelta de un pulso de luz. La luz del láser se difracta en numerosos puntos donde se podría obtener la información completa de la escena, pero el Laser no emite pulsos y no está modulado.

**FIGURA 19: FUNCIONAMIENTO KINECT**



Fuente: [http://www.academia.edu/4437415/Especificaciones\\_Kinect](http://www.academia.edu/4437415/Especificaciones_Kinect)

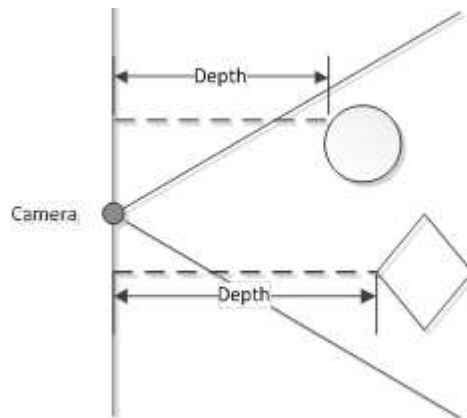
Al emitir el láser, la luz infrarroja se divide formando un patrón de puntos, al parecer aleatorios que podría tratarse siempre del mismo y ser constante. Se desconoce la razón por lo que tiene estas características, podría tratarse de secuencias derivadas del difractor empleado o formar parte de una serie de pistas que podrían utilizarse para algoritmos de correlación cruzada o triangulación activa.

#### **2.3.4. Técnica de Profundidad**

Las imágenes de profundidad constituyen un pilar fundamental en este proyecto, dejando claro el manejo y en que consiste su funcionamiento para la visualización de las imágenes.

Mediante el dispositivo se puede generar imágenes tridimensionales que reconoce los objetos que serán medidos a través del láser de luz infrarroja. El láser infrarrojo proyecta un cuadro de puntos no visibles para el ojo humano que se reflejan al chocar con los objetos siendo captado por la cámara infrarroja y procesado por el chip interno del Kinect (ver Figura 20).

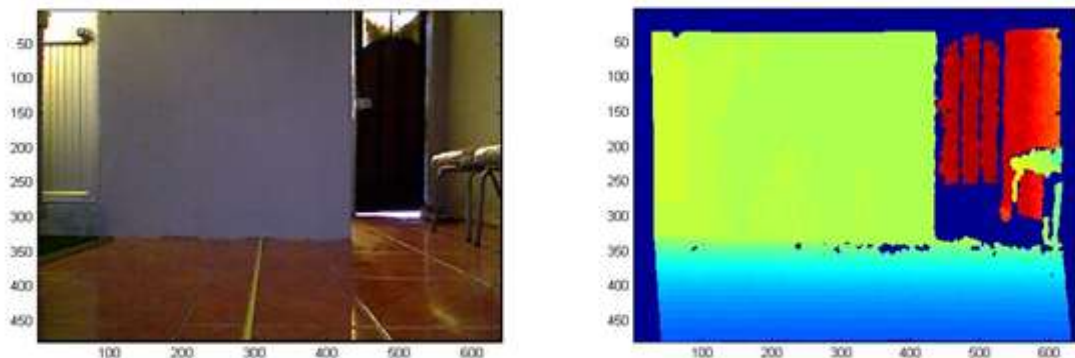
**FIGURA 20: PROFUNDIDAD KINECT**



**Fuente:** <http://msdn.microsoft.com/en-us/library/hh438997.aspx>

A través del circuito integrado analiza el patrón de puntos proyectado que es reflejado por el objeto el cual recibe la el sensor infrarrojo para generar la imagen del objeto como se muestra en la figura 21.

**FIGURA 21: IMAGEN DE PROFUNDIDAD**



**Fuente:** Álvarez R., Shagñay S.

La profundidad se obtiene mediante un cuadro de pixeles en una banda máxima de 30 imágenes por segundo. Se sabe que cada pixel está dado por dos bytes o 16 bits que representa la distancia del sensor al objeto. Para representar la ubicación de un pixel en la imagen de profundidad partimos de la distancia de la cámara hasta el objeto dada en milímetros para las coordenadas en los planos (x, y).

Al ser cero el valor del pixel significa que el sensor no obtiene la profundidad en esa región por la limitación del sensor. Si está demasiado cerca significa que el objeto fue detectado, pero está muy cerca del sensor para proporcionar una medición de distancia eficiente. Si está demasiado lejos significa que un objeto fue detectado, pero está muy lejos para poder medirse de forma eficiente y el valor desconocido significa que ningún objeto fue detectado.

## **CAPÍTULO III**

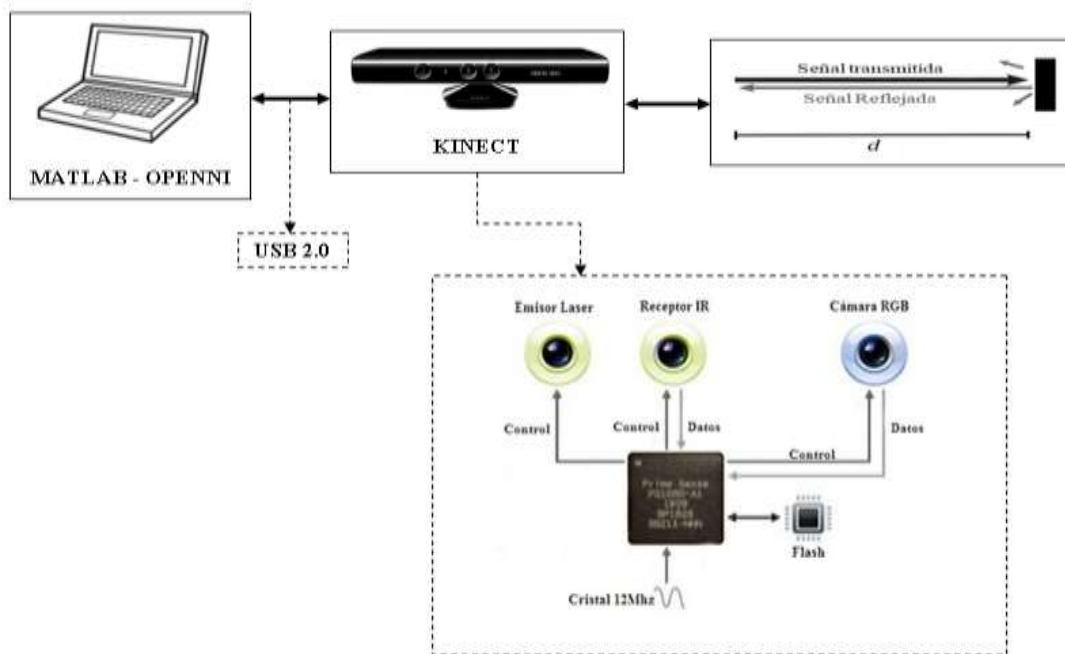
### **3. SISTEMA RADAR PARA RECONSTRUCCIÓN DE PERFILES TOPOGRAFICOS UTILIZANDO KINECT**

#### **3.1. SISTEMA RADAR UTILIZANDO KINECT**

Para el funcionamiento del sistema radar se utilizó el sensor Kinect que adaptado un PC y por medio de software se hizo posible la detección de objetos quedando el sistema como se muestra en la Figura 22.



**FIGURA 22: DIAGRAMA DE BLOQUES SISTEMA RADAR**



Fuente: Álvarez R., Shagñay S.

### 3.1.1. Señal Transmitida y Reflejada

Inicialmente se estudió la señal transmitida por el sensor Kinect que es una proyección de la luz infrarroja (laser) que incide sobre los objetos a través de un patrón de puntos, la señal rebota contra el objeto y es detectada por el receptor infrarrojo CMOS donde luego se estima la profundidad.

### 3.1.2. Kinect

La señal recibida por el sensor infrarrojo CMOS del Kinect es procesada por el circuito integrado de PrimerSense, el cual como se mencionó anteriormente se puede obtener una imagen en píxeles que representa la profundidad, esto a través de los datos de distancia calculados por el circuito integrado, estos datos se encuentran en milímetros.

Para poder conectar el Kinect con la computadora se requiere de un adaptador(ver Figura 23), el mismo que conecta el puerto de la salida del Kinect a un puerto USB 2.0, además alimenta al Kinect con una conexión eléctrica de 12v a 1.08 A.

**FIGURA 23:** ADAPTADOR KINECT



**Fuente:**[http://articulo.mercadolibre.com.ec/MEC-406847113-adaptador-ac-fuente-poder-sensor-kinect-xbox-360-tradicional-\\_JM](http://articulo.mercadolibre.com.ec/MEC-406847113-adaptador-ac-fuente-poder-sensor-kinect-xbox-360-tradicional-_JM)

Finalmente se hace uso de un PC que ayudará a la adquisición y visualización de datos, gracias a herramientas de software como Matlab, Open Ni y librerías que se menciona mas adelante.

### 3.1.3. Open NI

La empresa PrimerSense es quien desarrollo un framework que permite el manejo del sensor Kinect en múltiples plataformas. Permite comunicarse con sensores de audio, video y sensor de profundidad del Kinect, mientras tanto proporciona una API (Aplication Programing Interface) que conecta el hardware del equipo y las aplicaciones e interfaces del sistema operativo.

Además de proporcionar datos, este framework permite grabar la secuencia de imágenes a color en 3D en un archivo de extensión (.oni).

Open NI permite ser utilizada mediante cualquier sensor que posee chip PS1080 de PrimeSense como es:

- Microsoft Kinect

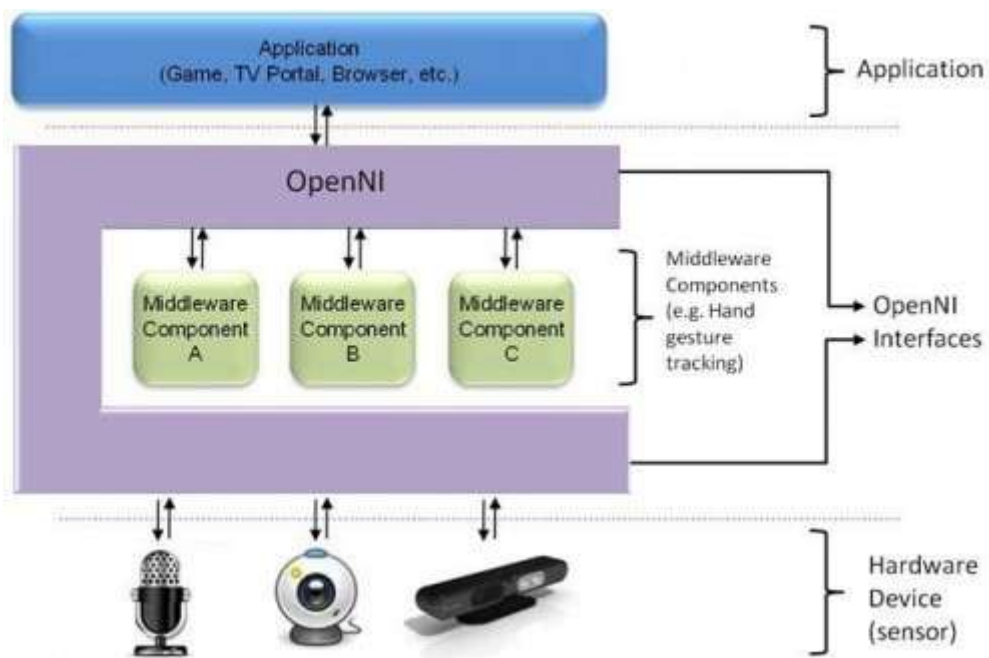
- ASUSXtion PRO(sin cámara RGB)
- ASUSXtion PRO Live(con RGB)
- PrimeSense PSDK

Siendo todos parecidos que contienen diferentes versiones.

### 3.1.3.1. Arquitectura

Para mejor entendimiento se hace énfasis en la arquitectura framework dispuesta como se muestra en la figura 24.

**FIGURA 24: ARQUITECTURA OPENNI**



*Fuente:* <http://animusproject.wix.com/web/apps/blog/openni-la-alternativa-open-source-para-hablar>

La capa de aplicación representa el desarrollo de aplicaciones como video juegos, navegación etc. En la capa de interfaz permite la conexión con los elementos que analizan los datos obtenidos por los sensores. La capa de hardware permite la comunicación con los sensores que obtienen los datos del usuario.

OpenNI permite acceder a dos tipos de componentes principalmente: sensores que extraen información del entorno, y elementos middleware que procesan y destilan la

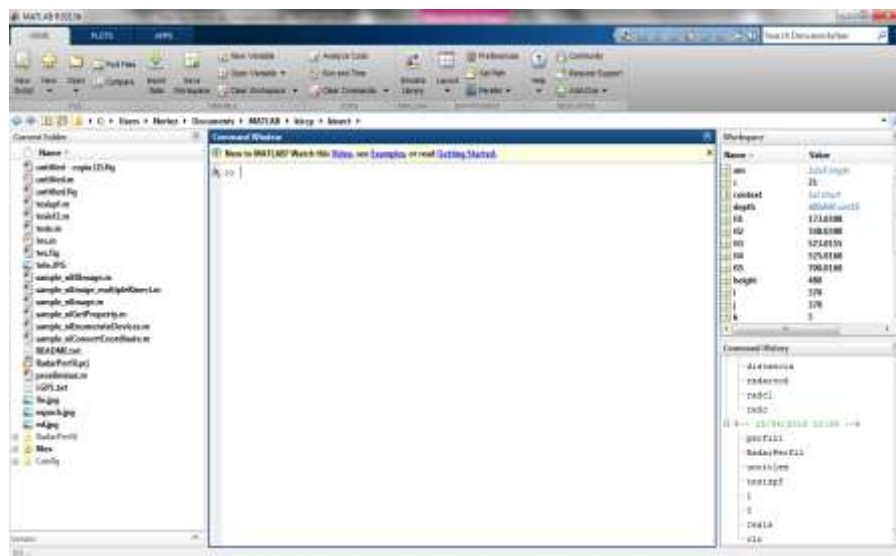
información obtenida por otros elementos para ofrecer y producir datos de más alto nivel. Estos componentes se denotan como módulos.

Se obtiene los datos 3D mediante el framework como son: imágenes del infrarrojo, las imágenes a color y los datos de audio.

### 3.1.4. MATLAB

Para el desarrollo de este proyecto se ha usado versión 2013b de MATLAB (ver Figura 25). Es un sistema interactivo basado en matrices para cálculos científicos y de ingeniería, se considera como un entorno matemático de simulación que puede utilizarse para modelar y analizar sistemas continuos, discretos, lineales y no lineales.

**FIGURA 25:** INTERFAZ MATLAB



Fuente: Álvarez R., Shagñay S.

#### 3.1.4.1. Características

Se conoce que Matlab es a la vez un entorno y también un lenguaje de programación, siendo uno de sus puntos fuertes el lenguaje Matlab que permite crear nuestras propias funciones y programas especiales en archivos .m y agruparlos en Toolbox (colección de archivos .m), obteniendo soluciones particulares en áreas de aplicaciones específicas.

### **3.1.4.2. Funcionamiento**

Matlab es capaz de procesar de modo secuencial una serie de comandos previamente definidos, obteniendo de forma inmediata los resultados. Los comandos pueden ser definidos por el usuario, donde ha de escribir la lista de comandos.

Para procesar los datos se debe seguir:

1. El usuario escribe expresiones en la ventana de comandos, o en un archivo de texto apropiado (archivo.m).
2. Tras la orden de ejecución, Matlab procesa la información.
3. Matlab escribe los resultados en la ventana de comandos y los gráficos en diferentes ventanas gráficas si los hubiere.

### **3.1.4.3. Archivos MEX**

Se pueden llamar rutinas C, C++ o subrutinas Fortran desde la línea de comandos de MATLAB como si fueran funciones incorporadas. Estos programas son llamados archivos binarios MEX, estas rutinas se vinculan dinámicamente y un intérprete de MATLAB las carga y ejecuta.

El archivo MEX contiene sólo una función o subrutina, y su nombre es mx\*nombre del archivo\*. Para llamar a un archivo MEX, se utiliza el nombre del archivo, sin la extensión de archivo.

El término MEX significa "MATLAB ejecutable" y tiene diferentes significados, como se muestra en la Tabla 2:

**TABLA 2: TÉRMINOS MEX**

| Termino MEX          | Definición  |
|----------------------|---|
| source MEX-file      | C, C++, o Fortran archivo de código fuente.   |
| binary MEX-file      | Subrutinas vinculadas dinámicamente que se ejecutan en MATLAB.  |
| MEX function library | MATLAB C/C++ y Fortran API librerías de referencia para realizar operaciones en el entorno de MATLAB. |
| mex build script     | Función de MATLAB para crear un archivo binario a partir de un archivo fuente.                        |

**Fuente:** [http://www.mathworks.com/help/matlab/matlab\\_external/introducing-mex-files.html](http://www.mathworks.com/help/matlab/matlab_external/introducing-mex-files.html)

#### **3.1.4.3.1. Kinect-Mex**

Para poder obtener datos del Kinect vamos a utilizar la colección de archivos Mex denominada Kinect-Mex.

Este proyecto es organizado por SourceForge.net., que es el mayor proveedor mundial de hosting para proyectos de desarrollo de software de código abierto y se encuentra disponible en la página de MathWorks. El equipo del proyecto lo describe como: Esta es una colección de MEX que permiten obtener imágenes (profundidad / RGB / IR) de Kinect en entorno Matlab. Son funciones de contenedor de APIs OpenNI. No se requieren toolboxes. La colección de archivos Mex son detalladas en la Tabla 3.

**TABLA 3: ARCHIVOS MEX**

| MEX file                         | Función  |
|----------------------------------|--|
| mxNiCreateContex                 | Conecta el Microsoft Kinect Hardware   |
| mxNiDeleteContext                | Cierra los objetos-Kinect que están conectados   |
| mxNiConvertProjectiveToRealWorld | Convierte píxeles (u, v) de una imagen de profundidad a profundidad [mm] para un mundo de coordenadas XYZ. |
| mxNiConvertRealWorldToProjective | Convierte profundidad [mm] para un mundo de coordenadas XYZ a píxeles (u, v) de una imagen de profundidad. |
| mxNiEnumerateDevices             | Obtiene la cantidad de dispositivos conectados   |
| mxNiGetProperty                  | Devuelve el valor de la propiedad  |
| mxNiImage                        | Obtiene la imagen RGB y de profundidad del Kinect  |
| mxNiIRImage                      | Obtiene la imagen IR y de profundidad del Kinect   |
| mxNiSetProperty                  | Establece un valor a una propiedad   |
| mxNiUpdateContext                | Espera que el Kinect arroje un video   |

Fuente: Álvarez R., Shagñay S.

### **3.2. APLICACIÓN PARA RECONSTRUCCIÓN DE PERFILES TOPOGRÁFICOS**

#### **3.2.1. Código en MATLAB**

A continuación se menciona las partes más importantes del código MATLAB que ayudaron a obtener datos y reconstruir el perfil.

En primer lugar se añade la dirección de la carpeta donde se encuentran los archivos Mex a utilizar, que es recomendable ubicar la carpeta en la que se genera el programa.

```
addpath('./Mex');
```

Luego se conecta el sensor Kinect con el programa.

```
context = mxNiCreateContext('Config/SamplesConfig.xml');
```

Se obtienen los datos de profundidad.

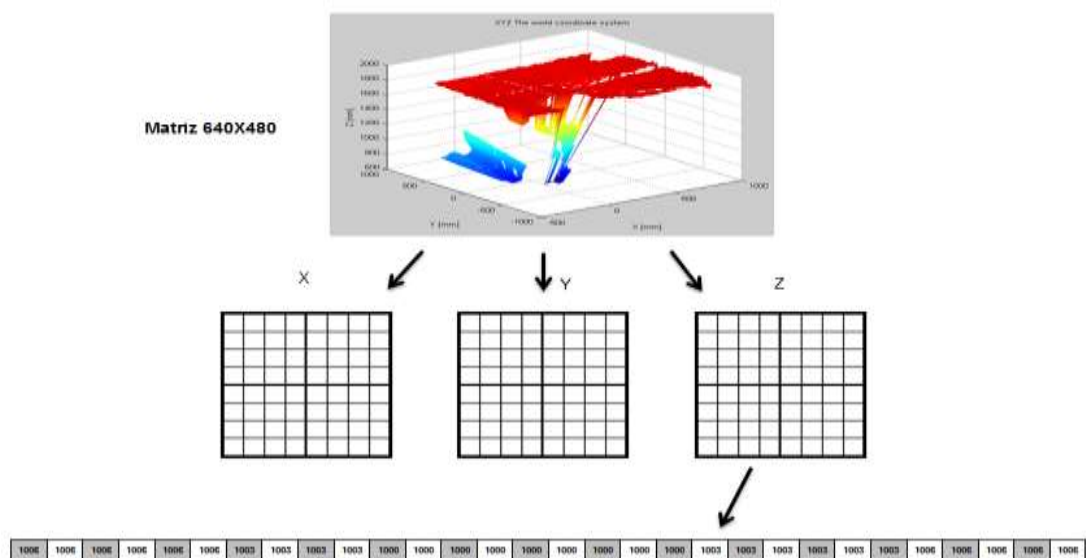
```
[rgb, depth] = mxNiImage(context);
```

Se llena la matriz `real_xyz` que contendrá los datos de profundidad.

```
real_XYZ = mxNiConvertProjectiveToRealWorld(context, depth);
```

Al momento de obtener la matriz `real_xyz` se grafica y se observa que la grafica en tres dimensiones posee áreas inconsistentes y no representa de forma fiel la superficie capturada por lo que se procede al análisis de sus datos, se descompone en sus tres elementos `y`, `x` y `z`, se localiza el elemento con los datos de profundidad en nuestro caso el elemento `z` y se procede a observar sus datos (ver Figura 26).

**FIGURA 26: DATOS KINECT**



Fuente: Álvarez R., Shagñay S.



Al observar los datos medidos a 1m de distancia (ver Tabla 4), se puede ver que los datos centrales son más precisos y los datos de los extremos mientras se alejan del centro aumentan y comienzan a volverse menos consistentes.

**TABLA 4: DATOS DE DISTANCIA**

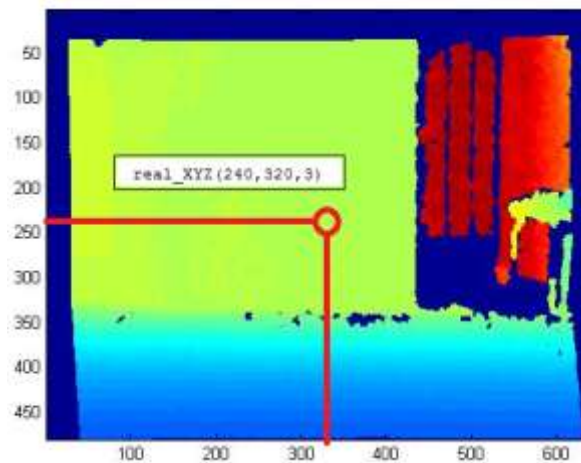
| <b>Datos Z</b> |
|----------------|
| 1003           |
| 1003           |
| 1003           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1000           |
| 1003           |
| 1003           |
| 1003           |

Fuente: Álvarez R., Shagñay S.

Por lo analizado, se considera más relevantes los datos centrales y se procede a tomar el dato intermedio, así como se muestra en la Figura 27.

real\_XYZ(240, 320, 3)

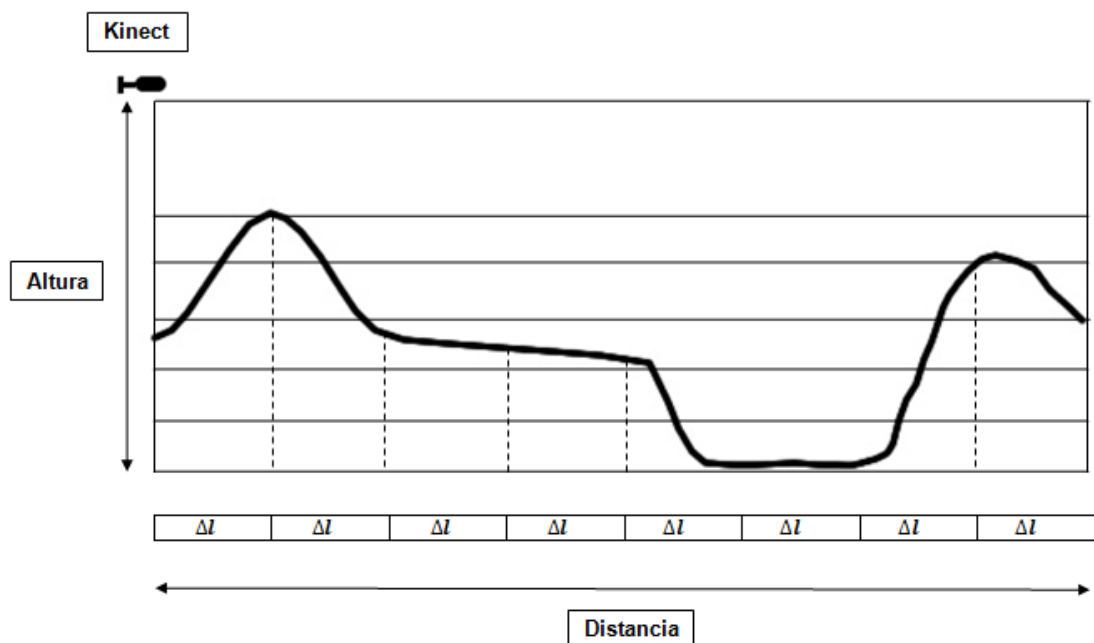
**FIGURA 27: SELECCIÓN DE DATOS KINECT**



Fuente: Álvarez R., Shagñay S.

Para poder reconstruir el perfil a partir de las medidas adquiridas, se determina la altura a la que se ubicara el Kinect con respecto al plano y la distancia a la que se tomara cada medición, como se observa en la Figura 28.

**FIGURA 28: RECONSTRUCCIÓN DEL PERFIL**



Fuente: Álvarez R., Shagñay S.

Los datos obtenidos de profundidad de la matriz `real_xyz` se almacenaran en un vector junto con los datos de distancia para poder ser graficados.

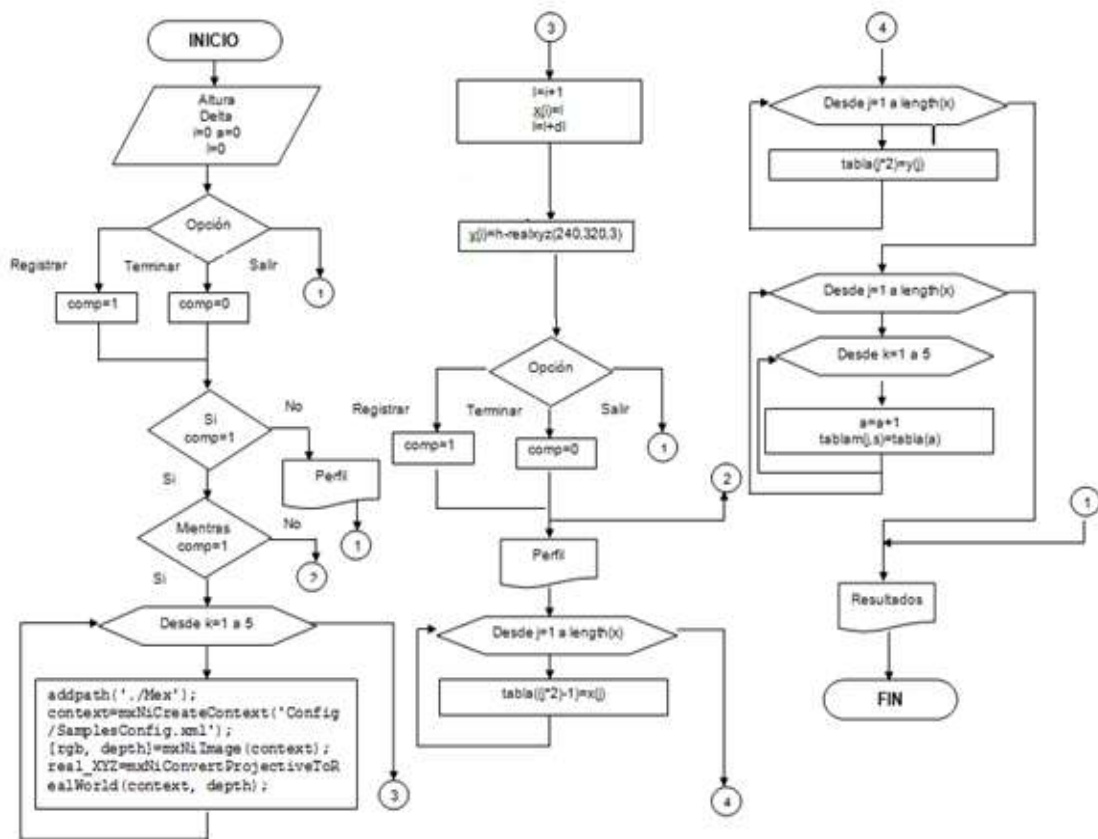
Como el perfil resultante se encuentra invertido se realiza una sustracción entre la altura a la que se encuentra el Kinect y el perfil obtenido como se muestra en el código.

```

real=(real_XYZ(240,320,3)/1000); %seleccionar datos perfil
if real>=h
    y(i)=0;
else
    y(i)=h-real;
end
x(i)=1;
    
```

A continuación en la Figura 29, se observa el flujograma del programa

**FIGURA 29: FLUJOGRAMA DEL PROGRAMA**



Fuente: Álvarez R., Shagñay S.

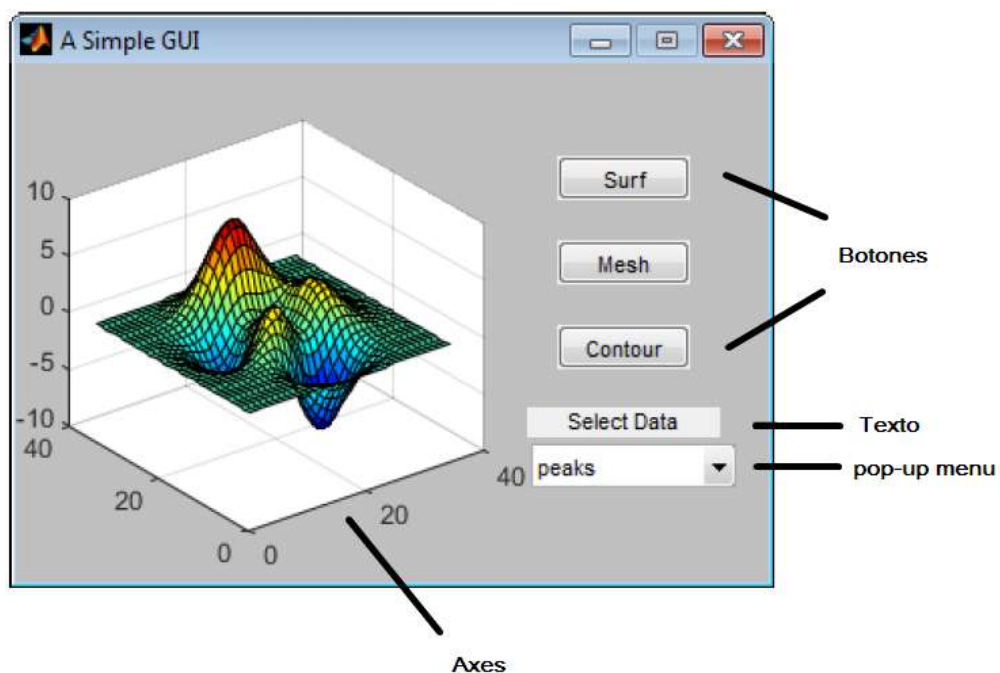
### 3.2.2. GUI en MATLAB

Para completar la aplicación se realizó una interfaz gráfica utilizando GUIDE de MATLAB.

GUI mejor conocida como interfaz gráfica de usuario, es una ventana grafica que contiene controles que permiten a un usuario realizar tareas interactivas. El usuario no tiene la necesidad de escribir un script o insertar líneas de comando para realizar la tarea.

Los componentes de una GUI pueden ser menús, barra de tarea, botones, listas entre otros (ver Figura 30). Una GUI creada en MATLAB también puede leer y escribir ciertos archivos así como comunicarse con otra GUI y mostrar datos como tablas o gráficos.

**FIGURA 30:** GUI CREADA EN MATLAB



**Fuente:** The MathWorks Inc. MATLAB Creating Graphical User Interfaces, USA, MathWorks, 2015, Pp. 1.

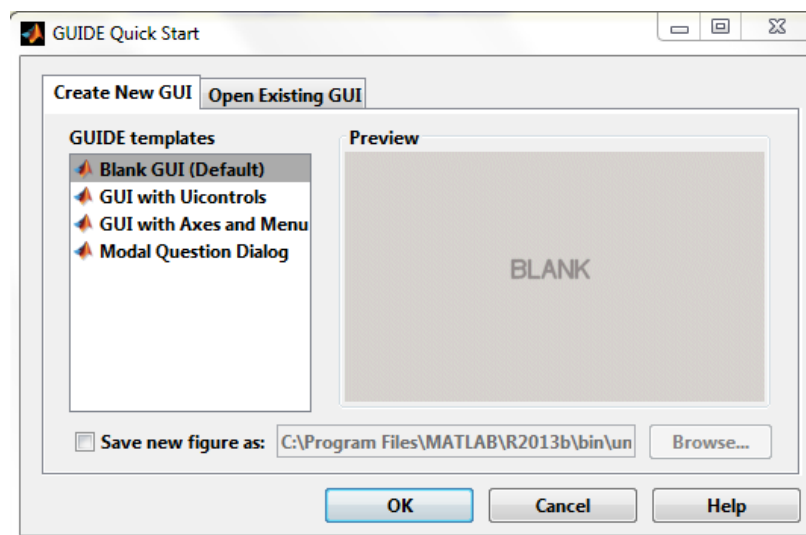
Típicamente, Una GUI espera que el usuario manipule un control entonces responde a cada opción en turno. Cada control y la misma GUI tienen una o más *callbacks* o devoluciones de llamada, llamadas así porque solicitan a MATLAB que realice una

tarea. Una determinada acción que realice el usuario como presionar un botón o desplazar el cursor sobre un componente dispara la ejecución de un callback. La GUI responde a estos eventos.

Este tipo de programación se refiere a menudo como programación orientada a eventos. En la programación orientada a eventos la ejecución de devolución de llamada es asíncrona, es decir, eventos externos al software pueden disparar la ejecución de llamada. En el caso de MATLAB GUI, la mayoría de los eventos son interacciones del usuario con la GUI, pero la GUI puede responder a otro tipo de eventos, así, por ejemplo, la creación de un archivo o la conexión de un dispositivo a la computadora.

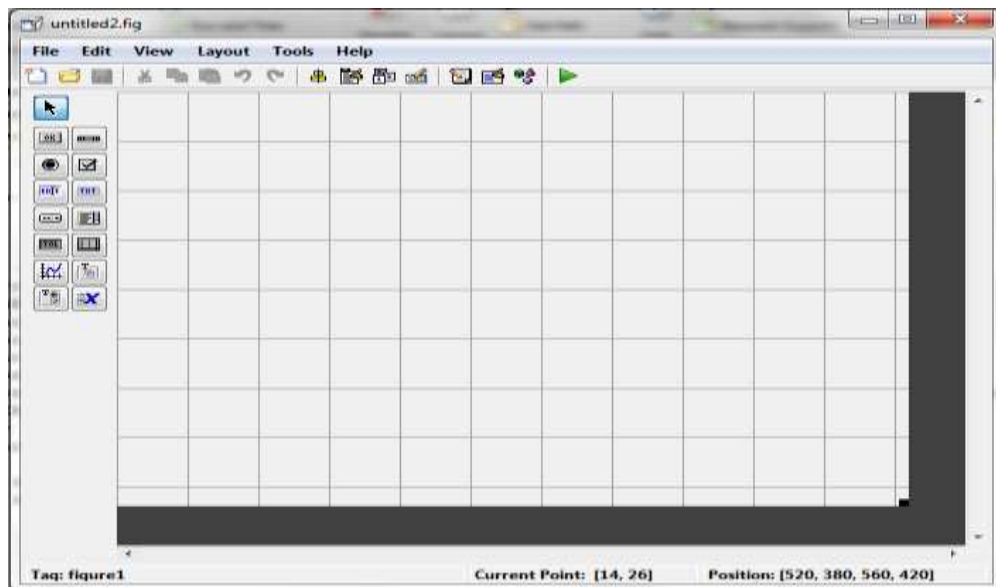
Al momento de crear un GUI en MATLAB (ver Figura 31) se utilizó el editor gráfico GUIDE en donde se puede seleccionar el tamaño, la posición y los componentes que se desea, como se puede ver en la Figura 32.

**FIGURA 31:** NUEVO GUI GUIDE MATLAB



**Fuente:** Álvarez R., Shagñay S.

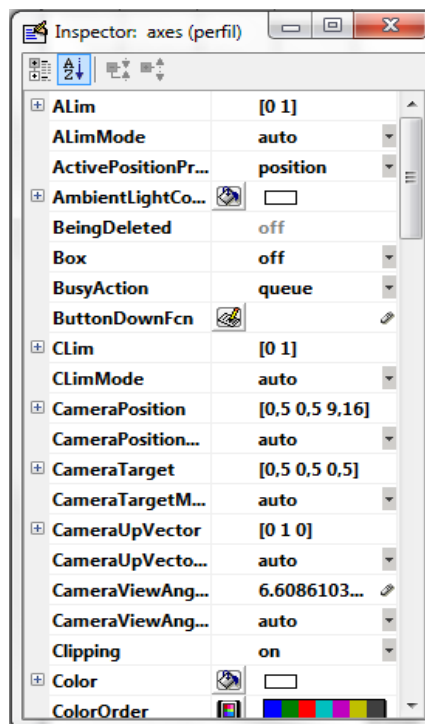
**FIGURA 32: VENTANA EDITOR GUIDE MATLAB**



Fuente: Álvarez R., Shagñay S.

Además se puede editar las propiedades de los componentes con un doble click sobre el mismo, donde se despliega la ventana mostrada en la Figura 33.

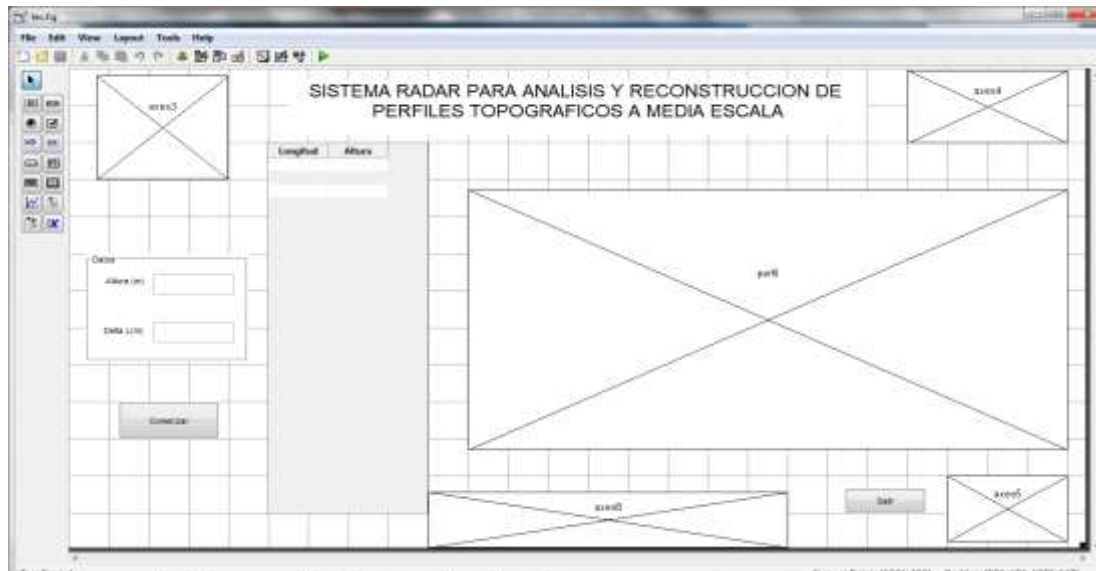
**FIGURA 33: PROPIEDADES DE LOS COMPONENTES**



Fuente: Álvarez R., Shagñay S.

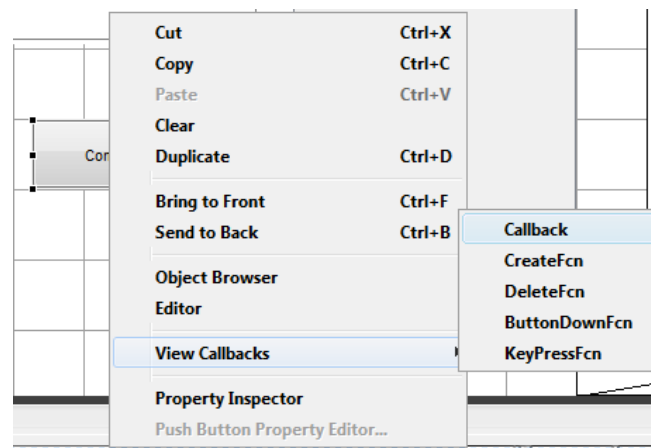
Luego de ubicar los componentes que se desea (ver Figura 34), y modificar sus propiedades se procede a editar sus callbacks mediante código, es en donde se edita el programa. Esto se realiza dando un click derecho sobre los componentes y seleccionando la opción Callback, como se observa en la Figura 35.

**FIGURA 34:** GUI CON SUS COMPONENTES



Fuente: Álvarez R., Shagñay S.

**FIGURA 35:** CALLBACK



Fuente: Álvarez R., Shagñay S.

```

function inicio_Callback(hObject, eventdata, handles)
% hObject    handle to inicio (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Get user input from GUI
% EDITAR CODIGO

```

Finalmente se obtiene la interfaz gráfica de usuario (ver Figura 36), donde se puede ingresar los datos de altura, desplazamiento y visualizar los resultados medidos en forma de perfil y datos.

**FIGURA 36: SISTEMA RADAR GUI**



Fuente: Álvarez R., Shagñay S.

El código completo del programa se encuentra en el anexo 4.



## CAPÍTULO IV

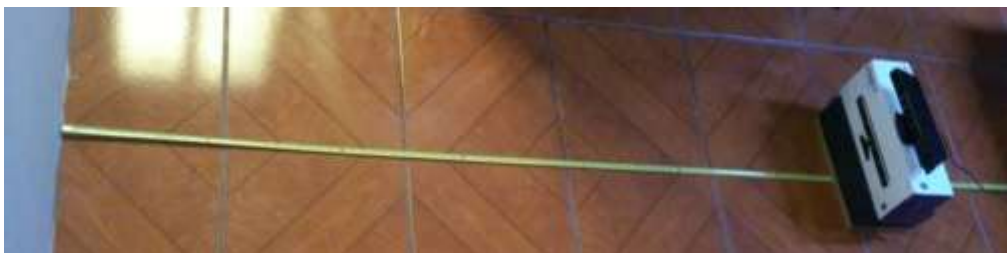
### 4. MEDICIONES Y RESULTADOS

#### 4.1. PRUEBAS DE DISTANCIA

##### 4.1.1. Distancia con Diferentes Materiales

Para la medición de distancia se consideró tres tipos de materiales cartón, aluminio y pared a distancias 1m, 2m, 3m y 4m de distancia, ubicando el sensor Kinect como se muestra en la Figura 37.

**FIGURA 37:** MEDICIÓN DE DISTANCIA



Fuente: Álvarez R., Shagñay S.

En la Tabla 5 se representan los resultados obtenidos.

**TABLA 5: DISTANCIAS**

|     | Pared | Aluminio | Cartón |
|-----|-------|----------|--------|
| 1 m | 1.000 | 1.000    | 1.000  |
| 2 m | 2.039 | 2.027    | 2.039  |
| 3 m | 3.101 | 3.073    | 3.101  |
| 4m  | 4.192 | 4.192    | 4.192  |

Fuente: Álvarez R., Shagñay S.

#### 4.1.2. Resolución y Frecuencia de Muestreo

Para obtener la resolución se tomó medidas se un plano que se encontraba a determinada distancia, y se lo desplazo progresivamente hasta notar un cambio en los valores medidos. Las medidas obtenidas se encuentran a continuación:

- **Distancia a 1 m**

**TABLA 6: DISTANCIA 1M**

| Real (m) | Medida (m) |
|----------|------------|
| 0.994    | 0.994      |
| 0.995    | 0.997      |
| 0.996    | 0.997      |
| 0.997    | 0.997      |
| 0.998    | 0.997      |
| 0.999    | 0.997      |
| 1        | 1.000      |
| 1.001    | 1.003      |
| 1.002    | 1.003      |
| 1.003    | 1.003      |
| 1.004    | 1.003      |
| 1.005    | 1.003      |
| 1.006    | 1.006      |

Fuente: Álvarez R., Shagñay S.

La Tabla 6 describe las mediciones que se realizaron en el entorno de un metro desplazándose cada milímetro de forma ascendente y descendente que permitirá darnos un aproximado de la precisión del dispositivo Kinect.

- **Distancia a 2 m**

**TABLA 7: DISTANCIA 2M**

| <b>Real (m)</b> | <b>Medida (m)</b> |
|-----------------|-------------------|
| 1.980           | 2.027             |
| 1.985           | 2027              |
| 1.990           | 2039              |
| 2               | 2039              |
| 2.05            | 2039              |
| 2.010           | 2039              |
| 2.015           | 2039              |
| 2.020           | 2051              |

**Fuente:** Álvarez R., Shagñay S.

De igual manera se realiza diferentes mediciones, pero en la Tabla 7 tenemos desplazamientos de 0.05 milímetros, tomando cuatro mediciones antes y después de los 2 metros.

- **Distancia a 3 m**

**TABLA 8: DISTANCIA 3M**

| <b>Real (m)</b> | <b>Medida (m)</b> |
|-----------------|-------------------|
| 2.975           | 3073              |
| 2.985           | 3101              |
| 2.990           | 3101              |
| 2.995           | 3101              |
| 3               | 3101              |

Continuara...

Continua:...

|       |      |
|-------|------|
| 3.005 | 3101 |
| 3.010 | 3101 |
| 3.015 | 3101 |
| 3.020 | 3101 |
| 3.025 | 3129 |

Fuente: Álvarez R., Shagñay S.

En la Tabla 8 se tomó mediciones antes y después de la medida establecida y presenta desplazamientos de 5 milímetros.

- **Distancia a 4 m**

**TABLA 9: DISTANCIA 4M**

| <b>Real (m)</b> | <b>Medida (m)</b> |
|-----------------|-------------------|
| 3.940           | 4.142             |
| 3.950           | 4.192             |
| 3.960           | 4.192             |
| 3.970           | 4.192             |
| 3.980           | 4.192             |
| 3.990           | 4.192             |
| 4               | 4.192             |
| 4.010           | 4.243             |

Fuente: Álvarez R., Shagñay S.

Las últimas mediciones son tomadas en intervalos de 0.10 metros. La Tabla 9 tiene medidas tomadas de 4 metros y descendiendo en 6 mediciones y una medición ascendente.

Como se observó los datos adquiridos a 1m son más precisos se tomó la resolución obtenida de 3mm para saber a qué frecuencia de muestreo trabaja nuestro sistema.

Como se menciona anteriormente en el análisis del sistema radar la distancia a la que se encuentra el objetivo está dada por:

$$2d = c * t \quad (4.1)$$

Y realizamos los siguientes calculos:

$$d_1 = c * \frac{t}{2} \quad (4.2)$$

$$t = \frac{2 * 1m}{3e^8 m/s} \quad (4.3)$$

$$t = 6.67e^{-9}s \quad (4.4)$$

$$d_2 = d_1 + \Delta d \quad (4.5)$$

$$d_1 + \Delta d = c * \frac{t_1}{2} \quad (4.6)$$

$$1.003 = c * \frac{t_1}{2} \quad (4.7)$$

$$t_1 = \frac{2 * 1.003}{3e^8 m/s} \quad (4.8)$$

$$t_1 = 6.69e^{-9}s \quad (4.9)$$

$$t_1 = \frac{2 * 1.003}{3e^8 m/s} \quad (4.10)$$

$$t_s = t_1 - t \quad (4.11)$$

$$t_s = 6.69e^{-9} - 6.67e^{-9}s \quad (4.12)$$

$$t_s = 3.33e^{-12}s \quad (4.13)$$

$$f_s = \frac{1}{t_s} \quad (4.14)$$

$$f_s = \frac{1}{3.33e^{-12}s} \quad (4.15)$$

$$f_s = 3e^{11} = 0.3 THz \quad (4.16)$$

### 4.1.3. Exactitud

Una vez realizado distintas mediciones a diferentes distancias es preciso definir la exactitud como la capacidad que posee un instrumento para acercarse al valor de la magnitud real.

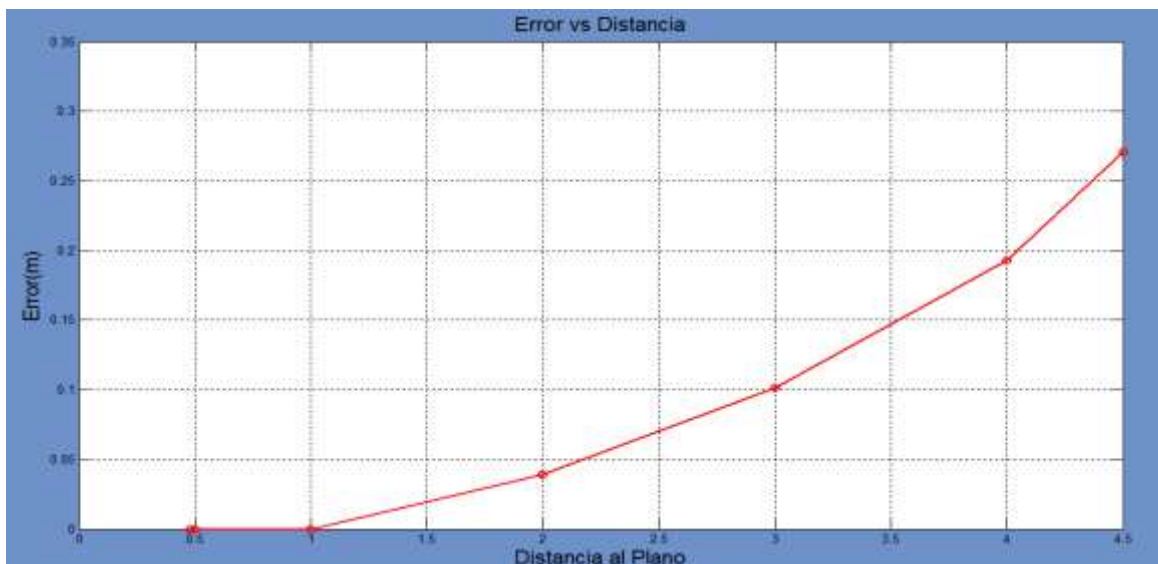
**FIGURA 38: ESCALA MEDIDA VS ESCALA REAL**



Fuente: Álvarez R., Shagñay S.

En la Figura 38 se observa que a medida que la distancia aumenta pasado el primer metro se pierde precisión en la medida y existe un error que va en incremento.

**FIGURA 39: ERROR VS DISTANCIA**



Fuente: Álvarez R., Shagñay S.

Tomando en cuenta el error que se produce se puede decir que de 0.50 a 1 metros de distancia las mediciones son bastante exactas, a 2 metros se encuentra un error de 3

centímetros que aún es aceptable, pero a partir de los 3 metros es superior a los tres centímetros (ver Figura 39). Para compensar este error se realiza una modificación al programa donde se lo compara con la distancia obtenida y se procede a su corrección.

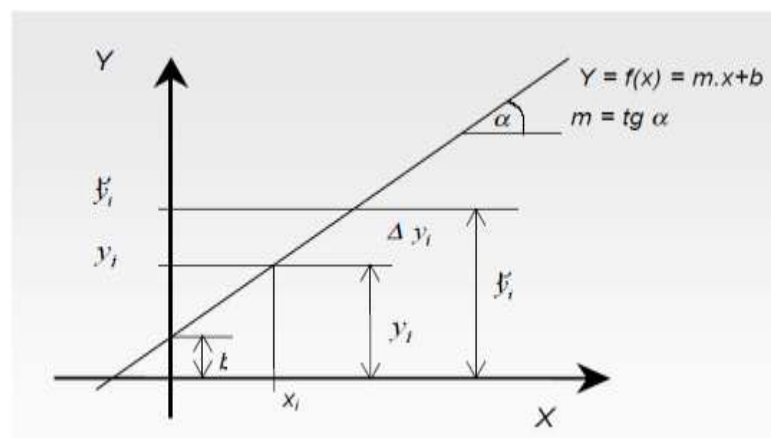
#### 4.1.4. Compensación de Error (Método de los Mínimos Cuadrados)

Los errores sistemáticos pueden deberse a defectos de calibración o construcción del dispositivo, puede también que al realizar la medición se cometerán errores al instante de ubicar el dispositivo o deberse a otros agentes externos. Es decir que cada medida tomada diferirá de la medida real en cierto valor.

Si se tiene dos series de magnitudes medidas que se encuentran relacionadas con alguna función lineal y una de las series de magnitudes se ve afectada por un error, se procede a trazar una recta con el objeto de que se aproxime a los  $n$  puntos medidos calculando los parámetros  $m$  y  $b$  de la recta, de tal manera que mantenga una mínima diferencia con los puntos.

La recta que satisface esta condición (ver Figura 40) es aquella que verifica que la sumatoria de las desviaciones elevadas al cuadrado es mínima. Dicho de otra manera la recta buscada es aquella que pasa por el centro geométrico de la nube de puntos y para la cual se observa una menor dispersión.

**FIGURA 40:** RECTA DE REGRESIÓN



Fuente: Álvarez R., Shagñay S.



$$m = \frac{n \sum_{i=1}^n (x_i * Y_i) - \sum_{i=1}^n x_i * \sum_{i=1}^n Y_i}{n \sum_{i=1}^n (x_i)^2 - (\sum_{i=1}^n x_i)^2} \quad (4.17)$$

$$b = \frac{\sum_{i=1}^n (x_i)^2 * \sum_{i=1}^n Y_i - \sum_{i=1}^n x_i * \sum_{i=1}^n (x_i * Y_i)}{n \sum_{i=1}^n (x_i)^2 - (\sum_{i=1}^n x_i)^2} \quad (4.18)$$

Donde:

$m$  = Pendiente de la recta de ajuste

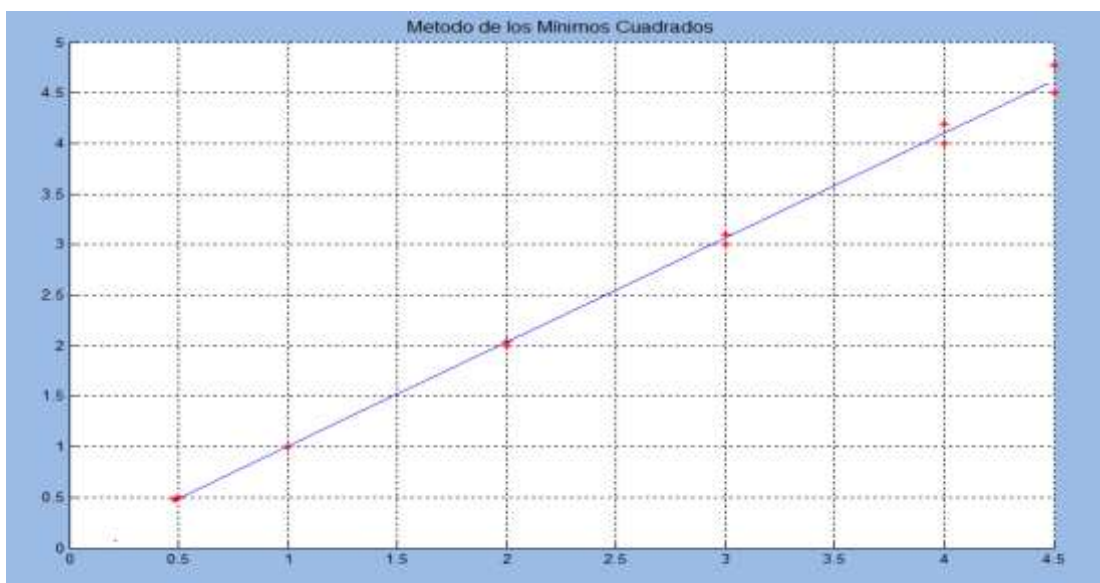
$b$  = Ordenada al origen.

En el caso que la ordenada sea cero la pendiente se simplifica:

$$m = \frac{\sum_{i=1}^n (x_i * Y_i)}{\sum_{i=1}^n (x_i)^2} \quad (4.19)$$

Al realizar la compensación del error de nuestro sistema se procede a obtener la curva de calibración con las medidas reales y las medidas obtenidas a través del método de los cuadrados mínimos, como se muestra en la Figura 41.

**FIGURA 41: MÉTODO DE LOS MÍNIMOS CUADRADOS**



Fuente: Álvarez R., Shagñay S.

Al realizar los cálculos por medio de MATLAB se obtiene una función lineal y los valores resultantes son:

$$m = 1.0314$$

$$b = -0.0263$$

La ecuación que se implementó en el programa para compensar el error está dada por:

$$Z = m * V_c + b \quad (4.20)$$

Donde:

$V_c$  = Valor corregido

$Z$  = Valor medido

Despejando:

$$V_c = \frac{Z - b}{m} \quad (4.21)$$

$$V_c = \frac{Z + 0.0263}{1.0314} \quad (4.22)$$

**TABLA 10: MEDIDAS COMPENSADAS**

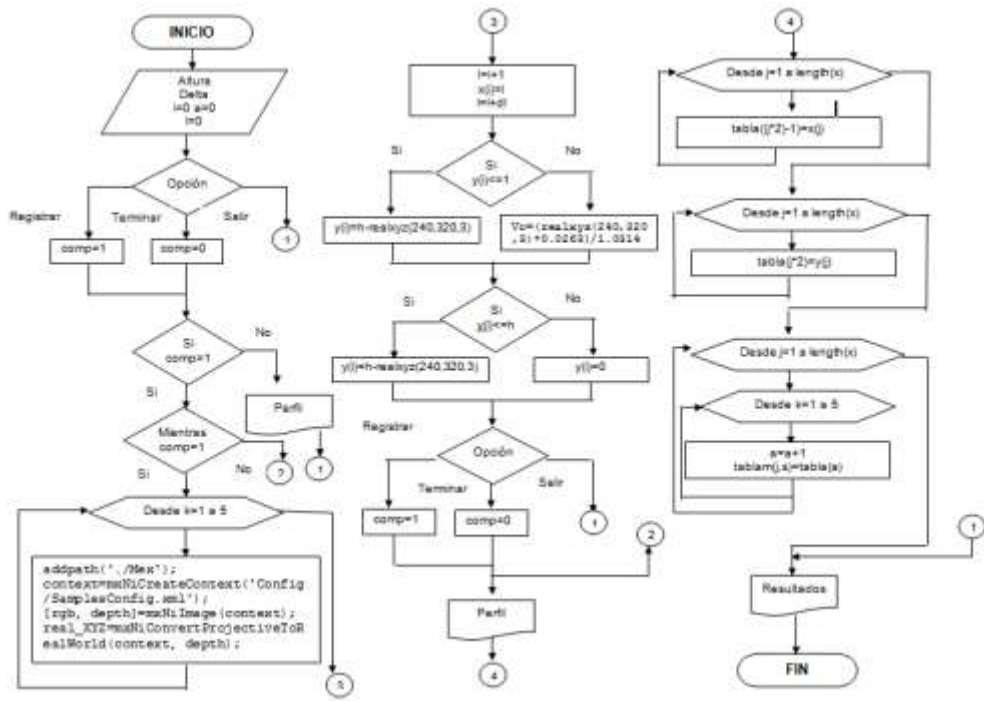
| Real (m) | Medido (m) | Compensado(m) |
|----------|------------|---------------|
| 0,48     | 0,48       | 0.4938        |
| 0,5      | 0,5        | 0.5103        |
| 1        | 1          | 0.9951        |
| 2        | 2,039      | 2.0024        |
| 3        | 3,101      | 3.0321        |
| 4        | 4,192      | 4.0899        |
| 4,5      | 4,771      | 4.6513        |

**Fuente:** Álvarez R., Shagñay S.

Al obtener los nuevos valores compensados se observa en la Tabla IV-6 que los valores menores a 1m pierden exactitud por lo que no son compensados en el programa, en

cambio los valores mayores a 1m que muestran mejoría si se compensan, quedando el flujograma del programa en la Figura 42; de la siguiente manera:

**FIGURA 42: FLUJOGRAMA (COMPENSACIÓN DE ERROR)**



Fuente: Álvarez R., Shagñay S.

## 4.2. PRUEBAS DE PERFIL

Al realizar las pruebas de perfil se tomó un plano en forma de escalón (ver Figura 43), para facilitar la comparación que se realizó posteriormente.

**FIGURA 43: PRUEBA DE PERFIL**



Fuente: Álvarez R., Shagñay S.

#### 4.2.1. Perfil a 1m

Se ubica el plano a la mitad de distancia de 1m, y se toma las medidas con dos desplazamientos el primero de 0.30 m y el segundo de 0.10 m. Teniendo como resultado la Figura 44.

**FIGURA 44:** PERFIL 1M DESPLAZAMIENTO 0.30 M



Fuente: Álvarez R., Shagñay S.

Disminuyendo el desplazamiento a 0.10m se obtiene la Figura 45.

**FIGURA 45:** PERFIL 1M DESPLAZAMIENTO 0.10 M



Fuente: Álvarez R., Shagñay S.

#### 4.2.2. Perfil a 2m

Así mismo se ubica el plano a la mitad de distancia de 2m, y se toma las medidas con dos desplazamientos el primero de 0.30 m y el segundo de 0.10 m. Teniendo como resultado la Figura 46.

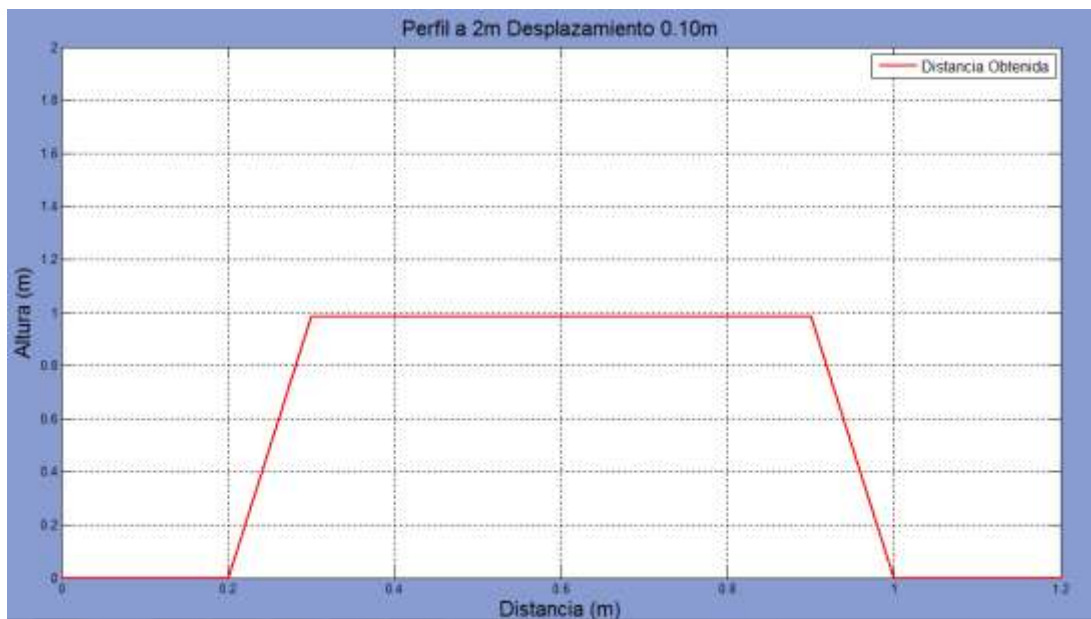
**FIGURA 46: PERFIL 2M DESPLAZAMIENTO 0.30 M**



Fuente: Álvarez R., Shagñay S.

De igual manera disminuyendo el desplazamiento a 0.10m se obtiene la Figura 47.

**FIGURA 47: PERFIL 2M DESPLAZAMIENTO 0.10 M**



Fuente: Álvarez R., Shagñay S.

Por último se realizó un perfil con dos planos a una distancia de 2m, el primer plano ubicado a 1.30m y el segundo plano a 1 m como se muestra en la Figura 48.

**FIGURA 48:** PERFIL DOS ELEMENTOS



Fuente: Álvarez R., Shagñay S.

Obteniéndose la Figura 49.

**FIGURA 49:** PERFIL 2M DOS ELEMENTOS DESPLAZAMIENTO 0.10 M

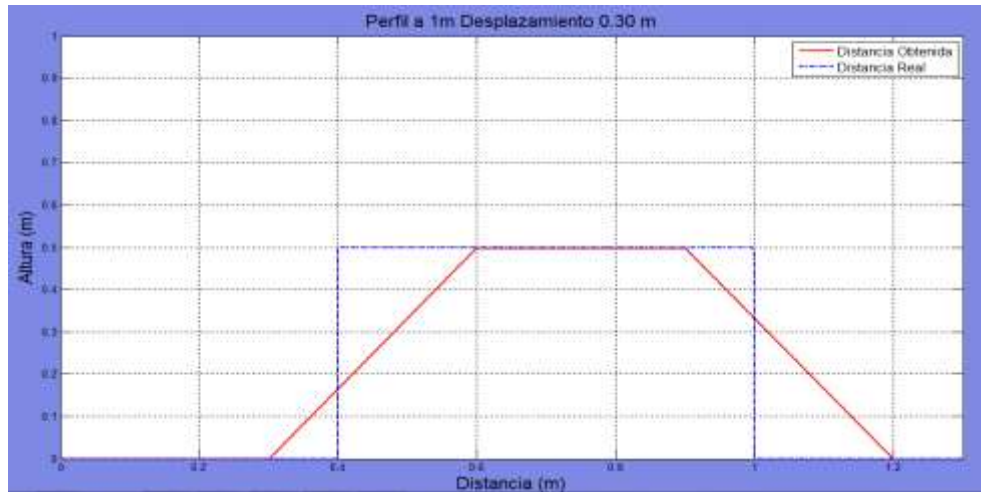


Fuente: Álvarez R., Shagñay S.

### 4.3. ANÁLISIS COMPARATIVO

#### 4.3.1. Perfil a 1m

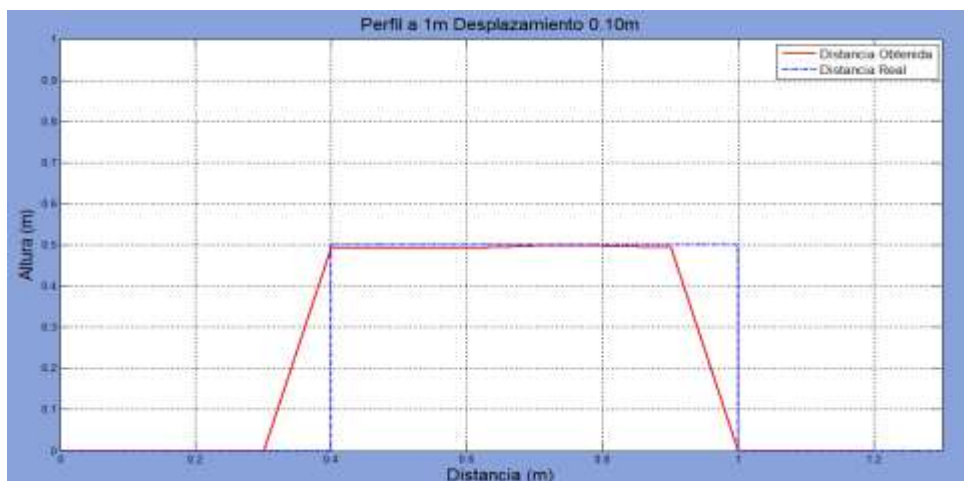
**FIGURA 50:** DISTANCIA REAL-OBTENIDA 1M 0.30M



Fuente: Álvarez R., Shagñay S.

A un desplazamiento de 0.30m como se observa en la Figura 50 no existe coincidencia en los bordes del plano y se observa más que solo se registra una parte.

**FIGURA 51:** DISTANCIA REAL-OBTENIDA 1M 0.10M



Fuente: Álvarez R., Shagñay S.

En intervalos de desplazamiento cortos, en este caso a 0.10m de desplazamiento existe una mejor coincidencia donde se puede identificar el borde del objeto a los 0.20m hasta aproximadamente 0.90m, luego se detecta una caída de distancia y no se observa el segundo borde (ver Figura 51).

### 4.3.2. Perfil a 2m

**FIGURA 52:** DISTANCIA REAL-OBTENIDA 2M 0.30M



Fuente: Álvarez R., Shagñay S.

Como se observa en la Figura 52 se ubicó un objeto a una distancia de 2m y medida en un intervalo de 0.30m, obteniendo como resultado una coincidencia en los bordes de los dos lados.

**FIGURA 53:** DISTANCIA REAL-OBTENIDA 2M 0.10M



Fuente: Álvarez R., Shagñay S.

En una segunda prueba de mediciones a 2m e intervalos de 0.10 m se asemeja a la forma del perfil del objeto obteniendo coincidencias de bordes y profundidad (ver Figura 53). Que permite constatar que el sensor Kinect trabaja de mejor manera a una distancia de 2m.



**FIGURA 54:** DISTANCIA REAL-OBTENIDA DOS ELEMENTOS 2M 0.10M



Fuente: Álvarez R., Shagñay S.

Observando las Figura 54 se puede constatar que existe una reconstrucción del perfil bastante acertada considerando los bordes rectos que posee. Al comparar los perfiles se logra apreciar que existe una coincidencia de perfil aproximada de 83.33% a un desplazamiento de 0.10 m y que a un menor desplazamiento  $\Delta l$  se obtiene una reconstrucción del perfil más precisa puesto que se adquiere una mayor cantidad de datos, incrementándose también el porcentaje de coincidencia.

## CONCLUSIONES

Al momento de analizar la señal de radar se puede decir que la Transformada Rápida de Fourier FFT es de gran utilidad ya que obtener el espectro de amplitud, observamos que los picos del mismo están relacionados con la posición del objetivo.

Mediante un algoritmo desarrollado en Matlab se ha implementado una aplicación capaz de obtener el perfil del plano u objeto que desee, donde el usuario final obtiene las distancia, imagen de profundidad y los puntos que forman dicho perfil.

Como se ha podido comprobar durante el transcurso del proyecto el sistema funciona correctamente. Sin embargo analizado el sensor Kinect se determinó que trabaja a una cierta distancia mínima de 0.48 m hasta 4.5 m en condiciones favorables, es decir que no exista luz solar (totalmente nublado) o por la noche.

Las mediciones tomadas a un metro no son tan precisas para obtener el perfil del objeto, a pesar que tiene desplazamiento de 0.10m y 0.30m no dan coincidencias de bordes del objeto medido. Sin embargo las mediciones a dos metros tienen mayor parecido al perfil del objeto con los mismos desplazamientos de un metro. Concluyendo que el sistema trabaja de mejor manera a dos metros y si menor es el desplazamiento de las mediciones mayor parecido tendrá el perfil del plano a medir.

El sensor Kinect refleja las múltiples aplicaciones que se le puede dar, desde el entretenimiento, medicina, hasta la geología. Tomándole al Kinect como el futuro que tiende a extender la interfaz de usuario-máquina. Este dispositivo permite a los desarrolladores crear distintas aplicaciones, pues anteriormente solo las grandes empresas o laboratorios de investigación manipulaban estos dispositivos.

## RECOMENDACIONES

El Kinect posee un rango de medición limitado que se debe considerar, para obtener el perfil topográfico de determinada zona se puede colocar el Kinect a una distancia que va desde los 0.50 m a los 4 m con respecto de los objetos, tomando en cuenta que a mayor distancia la exactitud y resolución comienzan a verse afectados.

El software desarrollado es más efectivo cuando las zonas que se van a medir no presentan áreas transparentes o reflectantes, puesto que no serán detectadas por el sensor de profundidad del Kinect.

Es recomendable que las mediciones sean en intervalos cortos y una distancia de dos metros, que el perfil será más real al tener mayor número de mediciones. Así mismo para adquirir los datos de un perfil con bordes rectos ubicar el proyector laser hacia la parte saliente del borde.

El uso de MATLAB para el desarrollo de aplicaciones de perfil es una opción excelente, ya que permite usar y modificar de manera sencilla los diferentes algoritmos, filtros y análisis binarios que forman parte del procesamiento digital de imágenes.

Como trabajo futuro el sistema podría ser aplicado a navegación autónoma de vehículos robóticos, además podría utilizarse para medir el desgaste en superficies glaciares.

## RESUMEN

En la siguiente investigación se desarrolló un sistema radar para análisis y reconstrucción de perfiles topográficos a media escala, para el grupo de investigación de electromagnetismo y microondas que posee la Facultad de Informática y Electrónica ubicada en la Escuela Superior Politécnica de Chimborazo.

Para el desarrollo del sistema se utilizó el sensor Kinect de Microsoft junto con el entorno de programación MATLAB: que puede utilizarse para simular modelar, analizar varios tipos de sistemas además, permite crear funciones y programas propios. El framework Open-NI permite manejar el sensor Kinect en múltiples plataformas, los archivos mex nos ayudan a adquirir los datos.

Al momento de poner a prueba el sistema a través de observación, mediciones y pruebas de perfil se obtienen datos de distancia, definiendo los parámetros del sistema, así como alcance, resolución, frecuencia de muestreo. Se optimiza el sistema compensando los errores medidos, mediante el método de cuadrados mínimos junto con la curva de calibración.

Se obtuvieron buenos resultados al momento de reconstruir el perfil topográfico teniendo una coincidencia de perfil aproximada de 83.33% desplazando a 0.10 m, considerando que adquiriendo más datos mejor será la reconstrucción del perfil incrementándose el porcentaje de coincidencia.

Se recomienda al grupo de investigación tomar en cuenta el rango de operación del sensor kinect, las condiciones de luz solar y tipo de superficie.

**Palabras Claves:** <SISTEMA RADAR> <PERFIL TOPOGRAFICO> <SENSOR KINECT> <SOFTWARE MATLAB> <FRAMEWORK OPEN-NI> <ELECTROMAGNETISMO> <LASER> <ARCHIVOS MEX>

## SUMMARY

The following research encloses the development of a Radar System for Analysis and Reconstruction of Medium-Scale Topographic Profiles for the research group of electromagnetism and microwave belonging to the faculty of informatics and electronics located at Escuela Superior Politécnica de Chimborazo.

In order to develop the system; it was used the Microsoft Kinect sensor with the MATLAB programming environment that can be used to simulate, model, analyze several types of systems, in addition, it allows to create functions and programs. The OPEN-NI framework makes possible to use the Kinect sensor in multiple platforms, mex-files help to collect data.

When testing the system by means of observation; measurements and testing profile; distance data are gathered defining the parameters of the system; such as scope, resolution, sampling frequency. The system is optimized by compensating the errors measured by Minimum Square Method with the calibration curve.

When reconstructing the topographic profile, the outcomes collected were good by showing a coincidence profile approximately 83.33 % displacing 0.10m, considering the fact that the data are collected the better will be the profile reconstruction increasing the coincidence percentage.

It is recommended that the research group take into account the operating range of the Kinect sensor, as well sunlight conditions and type of surface.

**Keywords:** <RADAR SYSTEM> <TOPOGRAPHIC PROFILES> <KINECT SENSOR> <MATLAB SOFTWARE> <OPEN-NI FRAMEWORK> <ELECTROMAGNETISM> <LASER> <MEX-FILES>

## GLOSARIO

**CARTOGRAFÍA:** representa la forma de la superficie terrestre en diferentes escalas.

**ESCALAS:** es la relación que existe entre las dimensiones reales y las dibujadas en planos, representando dimensiones o distancias de manera proporcional.

**FRAMEWORK:** es una estructura software configurable, para desarrollo de aplicaciones como: visión por ordenadores, desarrollo de juegos etc.

**GEODESIA:** estudia problemas relacionados con las figuras y dimensiones para el apoyo de levantamientos topográficos.

**GEORADAR:** instrumento de alta resolución para crear imágenes de subsuelo.

**GUI:** interfaz gráfica de usuario que permite la interacción de los sistemas a través de formas gráficas e imágenes.

**LASER:** dispositivo que produce luz mediante emisiones estimuladas de radiación.

**MIDDLEWARE:** software orientado a proporcionar conectividad para el funcionamiento de diferentes aplicaciones conectadas a través de una red.

**ONDA ELECTROMAGNETICA:** son ondas que no necesitan un medio material para propagarse como la luz visible.

**PERFIL:** líneas que marcan el límite de un objeto visto desde un punto determinado.

**PRECISIÓN:** refleja la proximidad de distintas medidas que mediante la suma de todos los errores absolutos representa un mismo resultado.

**RESOLUCIÓN:** es la menor variación que se puede detectar.

**TOPOGRAFÍA:** describe de manera detallada la superficie de un terreno, mediante líneas que unen un plano de referencia a una escala definida.

## BIBLIOGRAFÍA

1. **MELVIN W. L., SHEER J.A.** Principles of Modern Radar Applications Vol. III: Radar Applications. USA. SciTech Publishing. 2014. Pp. 1.
2. **MOORE H.** MATLAB para ingenieros. 1ra ed. México. Pearson Prentice Hall. 2007.
3. **RICHMOND R.D., CAIN S.C.** Direct-Detection LADAR Systems. USA. SPIE. 2010, Pp. 1.
4. **SZYMANCZYK O.** Historia de las Telecomunicaciones. Argentina. Dunken. 2013. Pp. 171-172-173.
5. **WOLF P.R., BRINKER R. C.** Topografía. 3<sup>a</sup> ed. México. Alfaomega. 1997. Pp.2.
6. **THE MATHWORKS INC.** MATLAB Creating Graphical User Interfaces, USA, MathWorks, 2015, pp. 2-4.
7. **APLICACIÓN DE LAS CÁMARAS 3D AL RECONOCIMIENTO DE ACTIVIDADES**  
<http://earchivo.uc3m.es/bitstream/handle/10016/16851/PFG%20%20Eric%20Magro%20Viforc.pdf?sequence=1>  
2015-02-17
8. **CHIRP**  
<http://en.wikipedia.org/wiki/Chirp>  
2015-01-28



## **9. CONCEPTOS BÁSICOS SOBRE OPENNI MÓDULOS**

[http://ocw.unia.es/cienciastecnologicas/tecnologiadelocio/materialesbasicosfolder/html/B1\\_UD02/mdulos.html/skinless\\_view](http://ocw.unia.es/cienciastecnologicas/tecnologiadelocio/materialesbasicosfolder/html/B1_UD02/mdulos.html/skinless_view)

2015-02-20

## **10. CORRELACIÓN CRUZADA**

[http://es.wikipedia.org/wiki/Correlaci%C3%B3n\\_cruzada](http://es.wikipedia.org/wiki/Correlaci%C3%B3n_cruzada)

2015-02-09

## **11. DESARROLLO DEL SISTEMA DE PERCEPCIÓN PARA UNA PLATAFORMA MÓVIL PARA ENTORNOS EXTERIORES**

[http://www.uned.es/ribim/volumenes/Vol8N3Nov\\_2004/V8N3A01%20Castejon.pdf](http://www.uned.es/ribim/volumenes/Vol8N3Nov_2004/V8N3A01%20Castejon.pdf)

2015-03-02

## **12. DESARROLLO Y EVALUACIÓN DE UN SISTEMA DE TELEOPERACIÓN DE ROBOTS BASADO EN RECONOCIMIENTO DE GESTOS USANDO EL SENSOR KINECT**

[http://oa.upm.es/26081/1/PFC\\_ALBERTO\\_GONZALEZ\\_DE\\_DIOS.pdf](http://oa.upm.es/26081/1/PFC_ALBERTO_GONZALEZ_DE_DIOS.pdf)

2015-03-02

## **13. ESCÁNER E IMPRESIÓN 3D**

<http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/Hardware/scanner3D/Esca%20ner3D.html>

2015-03-04

## **14. FUNDAMENTOS DE LA VISIÓN TRIDIMENSIONAL (3D)**

<http://www.escet.urjc.es/~visiona/tema6.pdf>

2015-02-09

## **15. GEOFORMAS**

<http://app.sni.gob.ec/snmlink/sni/PDOT/NIVEL%20NACIONAL/MAE/ECOSISTEMAS/DOCUMENTOS/GeofGeofo.pdf>

2014-11-27

## **16. GEOMETRÍA DE LA CARRETERA, PERFILES LONGITUDINAL Y TRANSVERSAL, RASANTE, SECCIONES TIPO**

[http://www.eclap.jcyl.es/web/jcyl/binarios/718/334/AUX\\_CAR\\_T05\\_FINAL.pdf?blobheader=application%2Fpdf%3Bcharset%3DUTF-8&blobheadername1=Cache-Control&blobheadername2=Expires&blobheadername3=Site&blobheadervalue1=no-store%2Cno-cache%2Cmust-revalidate&blobheadervalue2=0&blobheadervalue3=Portal\\_ECLAP&blobnocache=true](http://www.eclap.jcyl.es/web/jcyl/binarios/718/334/AUX_CAR_T05_FINAL.pdf?blobheader=application%2Fpdf%3Bcharset%3DUTF-8&blobheadername1=Cache-Control&blobheadername2=Expires&blobheadername3=Site&blobheadervalue1=no-store%2Cno-cache%2Cmust-revalidate&blobheadervalue2=0&blobheadervalue3=Portal_ECLAP&blobnocache=true)

Pp.13

2015-01-12

## **17. INTRODUCCIÓN A MATLAB**

<http://www.araba.ehu.es/depsi/jg/imatlab.pdf>

2015-02-13

## **18. INSTITUTO GEOGRÁFICO MILITAR DEL ECUADOR**

[http://www.igm.gob.ec/work/index.php?option=com\\_content&view=section&id=6&Itemid=3](http://www.igm.gob.ec/work/index.php?option=com_content&view=section&id=6&Itemid=3)

2014-11-27

## **19. INTRODUCCIÓN A LOS ARCHIVOS MEX**

[http://www.mathworks.com/help/matlab/matlab\\_external/introducing-mex-files.html](http://www.mathworks.com/help/matlab/matlab_external/introducing-mex-files.html)

2015-02-17

## **20. KINECT FOR MATLAB**

<http://www.mathworks.com/matlabcentral/linkexchange/links/2718-kinect-for-matlab>

2015-02-27

## **21. KINECT MATLAB**

<http://www.mathworks.com/matlabcentral/fileexchange/30242-kinect-matlab>

2015-02-27

## **22. LA ESCALA DE LOS MAPAS**

[132.248.9.9/libroe\\_2007/1048278/A06.pdf](http://132.248.9.9/libroe_2007/1048278/A06.pdf)

2015-01-14

## **23. MATLAB TUTOR**

<http://es.scribd.com/doc/218308118/Matlab-Tutor#scribd>

2015-02-13

## **24. MODULOS OPEN-NI**

[http://ocw.unia.es/ciencias-tecnologicas/tecnologia-del-ocio/materiales-basicos-folder/html/B1\\_UD02/mdulos.html/skinless\\_view](http://ocw.unia.es/ciencias-tecnologicas/tecnologia-del-ocio/materiales-basicos-folder/html/B1_UD02/mdulos.html/skinless_view)

2015-02-17

## **25. PRINCIPIOS DE LA TOPOGRAFÍA**

<http://ocw.utpl.edu.ec/ingenieria-civil/topografia-elemental/unidad-1-principios-de-topografia.pdf>

2014-11-27

## **26. SISTEMA DE APRENDIZAJE INTERACTIVO PARA EL ÁREA DEL IDIOMA INGLÉS CON EL SOPORTE DEL KINECT DE MICROSOFT**

<http://repositorio.espe.edu.ec/bitstream/21000/9251/1/T-ESPEL-SOF-007.pdf>

2015-02-10

## **27. SISTEMA DE RADARES ECUATORIANOS**

<http://www.desarrolloamazonico.gob.ec/ecuador-tiene-el-sistema-de-radaresmasmoderno-de-latinoamerica/>,

2014-11-27

## **28. TEORÍA DE ERRORES: CALIBRACIONES**

<http://www.aulavirtual->

[exactas.dyndns.org/claroline/backends/download.php?url=L1RQX05fMI9DYWxpYnJhY2lvbmVzLnBkZg%3D%3D&cidReset=true&cidReq=GUNIFDF](http://www.aulavirtual-exactas.dyndns.org/claroline/backends/download.php?url=L1RQX05fMI9DYWxpYnJhY2lvbmVzLnBkZg%3D%3D&cidReset=true&cidReq=GUNIFDF)

2015-04-08

# **ANEXOS**

## ANEXO 1: CÓDIGO RADAR SEÑAL NORMAL

```
clear all;
close all;
clc;

f=100e3;
n=100;
fmax=f;%frecuencia máxima

fs=(2*fmax); %frecuencia de muestreo
ts=1/fs; %tiempo de muestreo
tf=1000*ts;
t=0:ts:tf; %vector tiempo
Vt=2*cos(2*pi*f*t);%onda transmitida
Vr=[zeros(1,n),Vt(1:length(Vt)-n)]; %onda recibida
Vf=Vt.*Vr;%mul onda trans y onda recibida

k=length(Vf);

fvf=fftshift(fft(Vf)); %transformda de fourier

fg=[-fs/2:fs/(k-1):fs/2];

subplot(2,2,1)
plot(Vt)
grid on;
subplot(2,2,2)
plot(Vr)
grid on;
subplot(2,2,3)
plot(Vf)
grid on;
subplot(2,2,4)
plot(fg(floor(k/2):k),abs(fvf(floor(k/2):k))/floor(k/2))
grid on;
```

## ANEXO 2: CÓDIGO RADAR SEÑAL CHIRP

```
clear all;
close all;
clc;

R=100; %distancia
des=R*2;
c=3e8;
tdes=des/c;
f0=10e3; %frecuencia inicial

fase=0;%fase
dR=1; %precisión
fmax=c/(2*dR);%frecuencia máxima
f1=fmax;%frecuencia final
fs=2*fmax; %frecuencia de muestreo
display(fs);
ts=1/fs; %tiempo de muestreo
n=tdes/ts;
tf=(R*4)/c;%tiempo de muestreo
t=0:ts:tf; %vector tiempo
t0=t(1);
T=tf-t0;
k=(f1-f0)/T;
Vt=2*cos(2*pi*(k/2*t+f0).*t+fase);%señal transmitida

Vr=[zeros(1,n),Vt(1:length(Vt)-n)]; %señal recibida
Vf=Vt.*Vr;%mul señal trans y recibida

fvf=fftshift(fft(Vf)); %transformada de fourier

m=length(fvf);
h=round(m/2);
display(m);
display(h);
df=fs/(h);
dr=c/(h*df);
step=dr;
rg(1)=0;

for i=2:h
    rg(i)=dr;
    dr=dr+step;
end

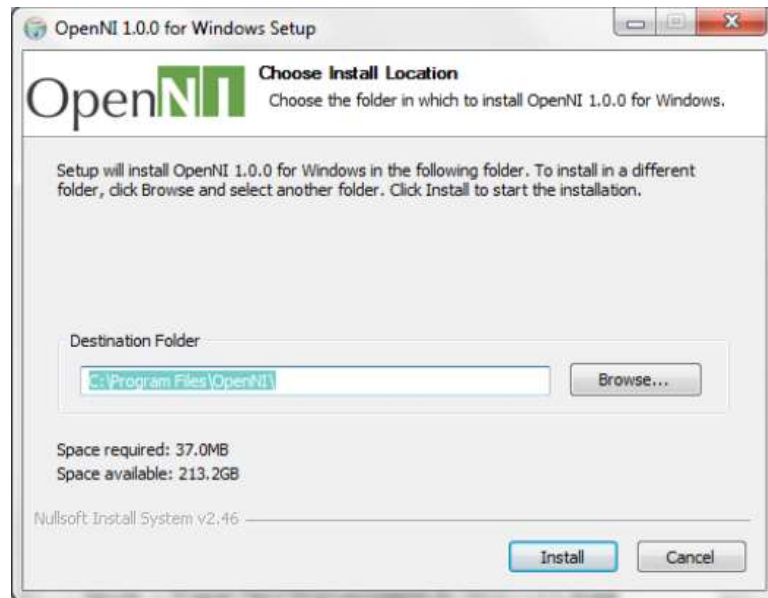
x=length(rg);
display(x);
p=get(0,'ScreenSize');
set(gcf,'Position',p);
subplot(2,2,1)
plot(Vt)
grid on;
subplot(2,2,2)
plot(Vr)
grid on;
subplot(2,2,3)
plot(Vf)
```

```
grid on;  
subplot(2,2,4)  
plot(rg,abs(fvf(round(m/2):m))/round(m/2));  
grid on
```



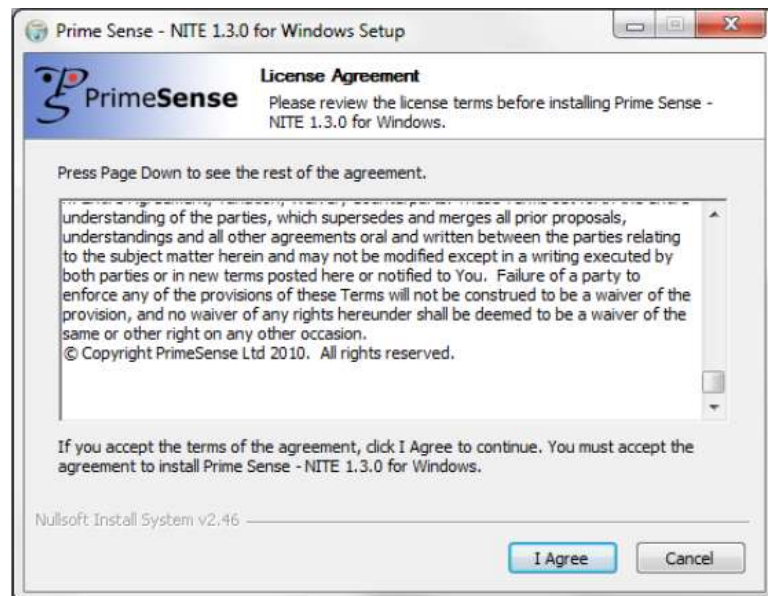
## ANEXO 3: INSTALACIÓN OPEN-NI

- **Instalar OpenNiv1.0.0.25**



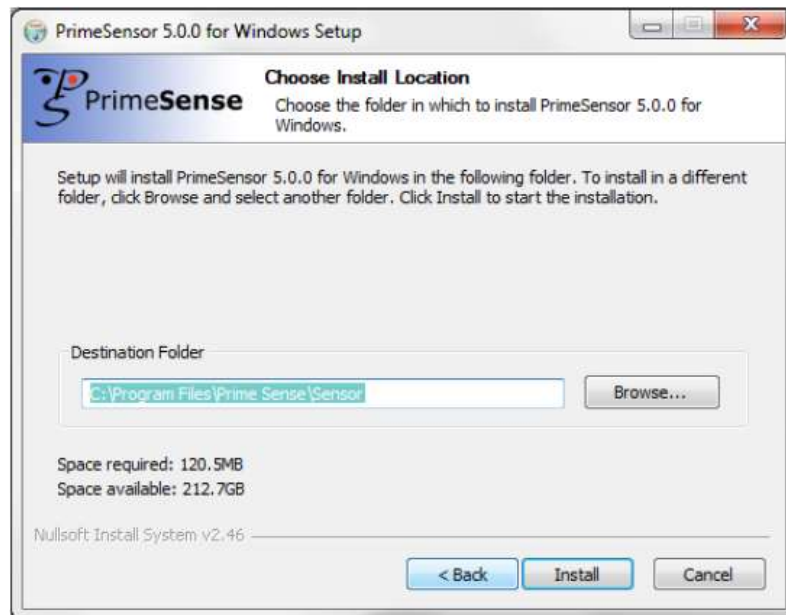
Fuente: Álvarez R., Shagñay S.

- **Instalar NITE v1.3.0.18**



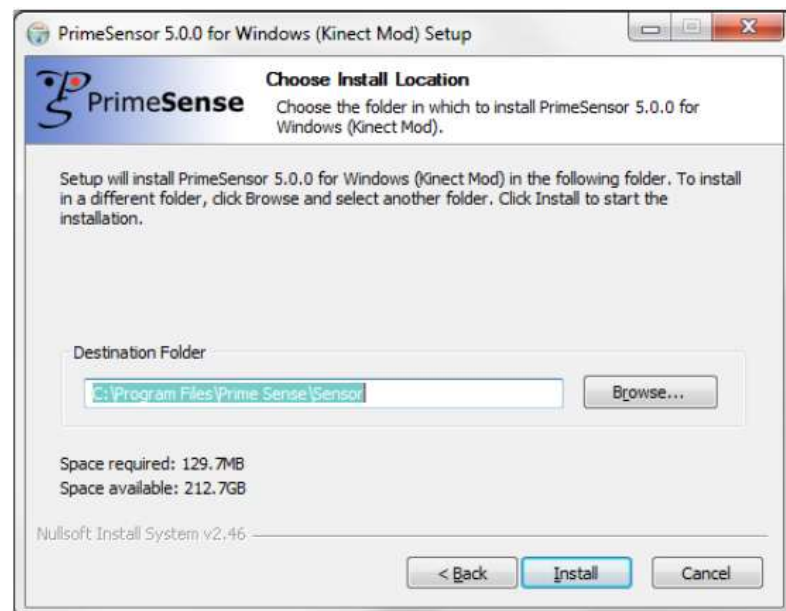
Fuente: Álvarez R., Shagñay S.

- **Instalar SensorKinect-Win32-5.0.0.exe**



Fuente: Álvarez R., Shagñay S.

- **Instalar SensorKinect(Mod)-Win32-5.0.0.exe**



Fuente: Álvarez R., Shagñay S.

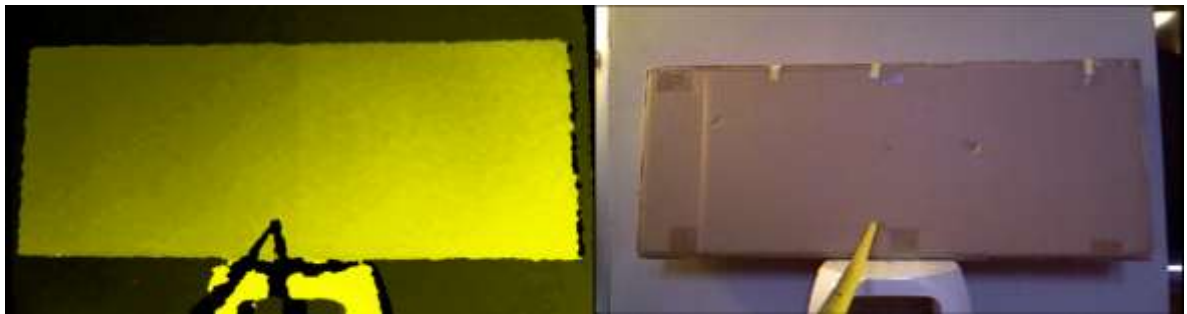
Si aparece un mensaje de alerta de Windows dar click sobre la opción instalar este software de controlador de todas maneras.



Fuente: Álvarez R., Shagñay S.

- Para comprobar su funcionamiento ejecutar el ejemplo ubicado en:

C:\Program Files\OpenNI\Samples\Bin\Release\NIViewer.exe



Fuente: Álvarez R., Shagñay S.

## ANEXO 4: CÓDIGO SISTEMA RADAR COMPLETO

```
function varargout = untitled(varargin)
% UNTITLED MATLAB code for untitled.fig
%     UNTITLED, by itself, creates a new UNTITLED or raises the
existing
%     singleton*.
%
%     H = UNTITLED returns the handle to a new UNTITLED or the handle
to
%     the existing singleton*.
%
%     UNTITLED('CALLBACK',hObject,eventData,handles,...) calls the
local
%     function named CALLBACK in UNTITLED.M with the given input
arguments.
%
%     UNTITLED('Property','Value',...) creates a new UNTITLED or
raises the
%     existing singleton*. Starting from the left, property value
pairs are
%     applied to the GUI before untitled_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to untitled_OpeningFcn via
varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help untitled

% Last Modified by GUIDE v2.5 29-Mar-2015 21:00:37

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @untitled_OpeningFcn, ...
                  'gui_OutputFcn',  @untitled_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

```

% --- Executes just before untitled is made visible.
function untitled_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to untitled (see VARARGIN)

% Choose default command line output for untitled
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
%imshow('esepoch.jpg','Parent',handles.axes3)
%
axes(handles.axes3)
background=imread('esepoch.jpg');
axis off;
imshow(background);
%
axes(handles.axes4)
background=imread('fie.jpg');
axis off;
imshow(background);
%
axes(handles.axes5)
background=imread('tele.jpg');
axis off;
imshow(background);
%
axes(handles.axes8)
background=imread('ed.jpg');
axis off;
imshow(background);

% UIWAIT makes untitled wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = untitled_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

function altura_Callback(hObject, eventdata, handles)
% hObject    handle to altura (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of altura as text
%        str2double(get(hObject,'String')) returns contents of altura
%        as a double

```

```

h = str2double(get(hObject,'String'));

if isnan(h) || ~isreal(h)
    set(handles.inicio,'String','Cannot plot f1')
    set(handles.inicio,'Enable','off')
    uicontrol(hObject)
else
    % Enable the Plot button with its original name
    set(handles.inicio,'String','Inicio')
    set(handles.inicio,'Enable','on')
end

% --- Executes during object creation, after setting all properties.
function altura_CreateFcn(hObject, eventdata, handles)
% hObject    handle to altura (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function delta_Callback(hObject, eventdata, handles)
% hObject    handle to delta (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of delta as text
%       str2double(get(hObject,'String')) returns contents of delta
as a double

dl = str2double(get(hObject,'String'));

if isnan(dl) || ~isreal(dl)
    set(handles.inicio,'String','Cannot plot f1')
    set(handles.inicio,'Enable','off')
    uicontrol(hObject)
else
    set(handles.inicio,'String','Inicio')
    set(handles.inicio,'Enable','on')
end

% --- Executes during object creation, after setting all properties.
function delta_CreateFcn(hObject, eventdata, handles)
% hObject    handle to delta (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.

```

```

% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in inicio.
function inicio_Callback(hObject, eventdata, handles)
% hObject handle to inicio (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Get user input from GUI

addpath('./Mex');
h = str2double(get(handles.altura,'String'));
dl = str2double(get(handles.delta,'String'));

choice = questdlg('Registrar primer Dato', ...
    'Opciones', ...
    'Registrar','Terminar','Salir','Salir');

switch choice
case 'Registrar'
    comp = 1;
case 'Terminar'
    comp = 0;
case 'Salir'
    close all;
end

i=0;
l=0;

if comp==1
    axes(handles.sencam)
    axis off;
    while comp==1

        context = mxNiCreateContext('Config/SamplesConfig.xml'); %%
        crear contexto con archivo xml Conecta el Microsoft Kinect Hardware
        width = 640; height = 480; %% inicializar figura
        fil = imagesc(zeros(height,width,'uint16'));% figura de
        profundidad

        for k=1:5
            option.adjust_view_point = true; %alinear profundidad
en rgb
            [rgb, depth] = mxNiImage(context); % Adquirir rgb y
            profundidad mxNiUpdateContext(context, option);
            real_XYZ = mxNiConvertProjectiveToRealWorld(context,
            depth); % convertir uvZ[pix,pix,mm] a XYZ[mm,mm,mm]
            set(fil, 'CData', depth); % actualizar figura
            tmp = real_XYZ(:);% actualizar figura XYZ
            tmp(find(real_XYZ==0)) = NaN;
            tmp = reshape(tmp,[height width 3]);

        end

        handles.sencam=fil; %muestra figura de profundidad
    end
end

```

```

mxNiDeleteContext(context); %% borrar el objeto de contexto
i=i+1;
real=(real_XYZ(240,320,3)/1000); %seleccionar datos perfil
if real<=1
    y(i)=h-real;
else
Vc=(real+0.0263)/1.0314;
if Vc>=h
    y(i)=0;
else
    y(i)=h-Vc;
end
end
x(i)=1;
l=l+dl;

choice = questdlg('Dato Registrado Registrar Siguiente Dato',
...
    'Opciones', ...
    'Registrar','Terminar','Salir','Salir');
switch choice
case 'Registrar'
    comp = 1;
case 'Terminar'
    comp = 0;
case 'Salir'
    close all;
end

end

axes(handles.perfil) %visualizar perfil
plot(handles.perfil,x,y)
grid on;
axis([0 l 0 h]);

b=length(x);
for j=1:b
    tabla((j*2)-1)=x(j);
end

for j=1:b
    tabla(2*j)=y(j);
end
g=length(tabla);
a=0;
for j=1:b
    for s=1:2
        a=a+1;
        tablam(j,s)=tabla(a);
    end
end

set(handles.uitable1, 'data', tablam)% visualizar tabla de valores

else

di=msgbox('No se han registrado datos');

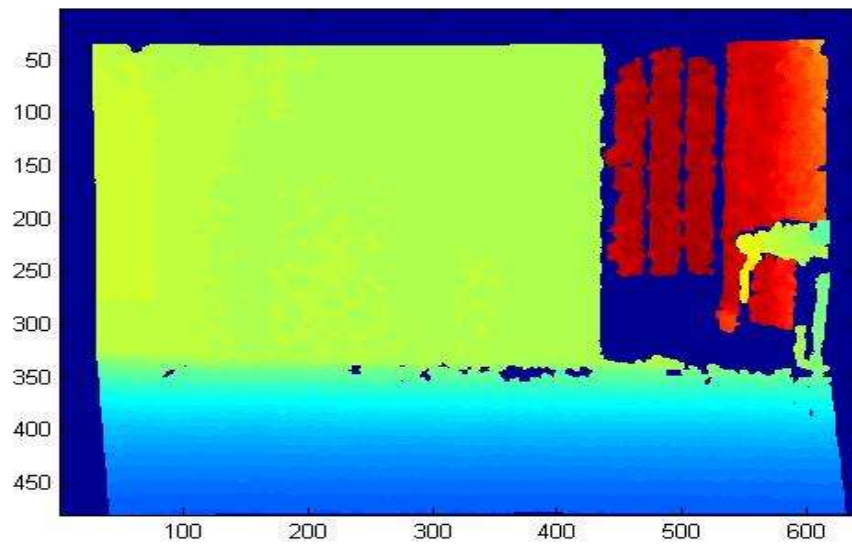
```



```
end
```

```
% --- Executes on button press in pushbutton2.  
function pushbutton2_Callback(hObject, eventdata, handles)  
% hObject    handle to pushbutton2 (see GCBO)  
% eventdata  reserved - to be defined in a future version of MATLAB  
% handles    structure with handles and user data (see GUIDATA)  
close all;
```

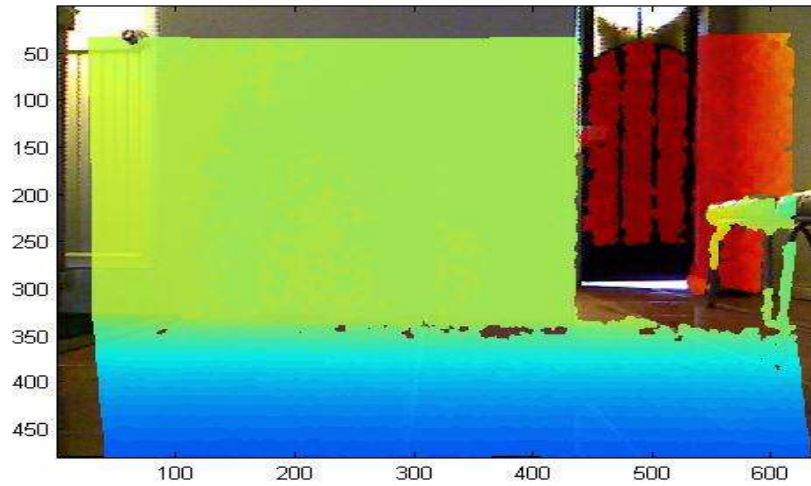
## ANEXO 5 : CAPTURAS



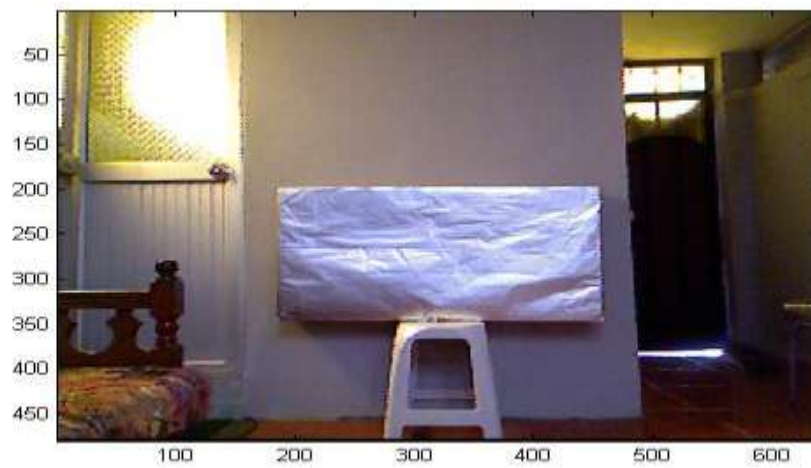
**Fuente:** Álvarez R., Shagñay S.



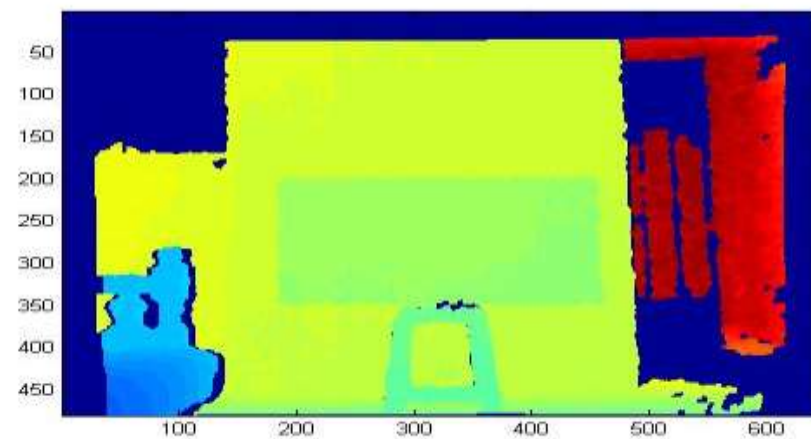
**Fuente:** Álvarez R., Shagñay S.



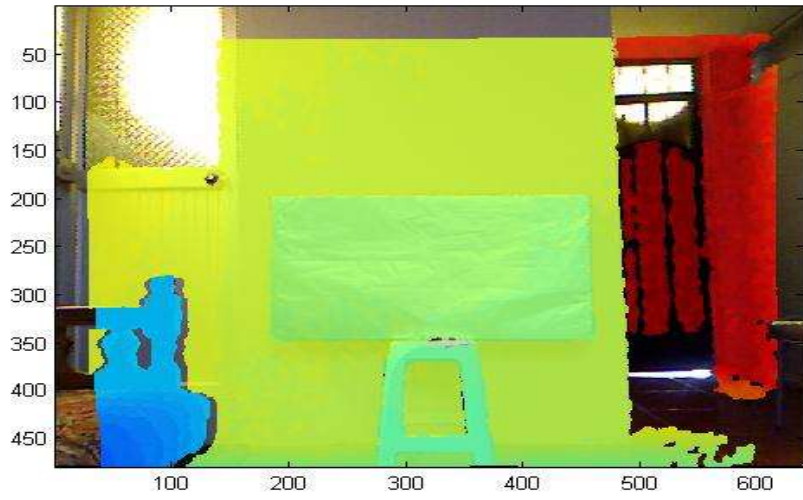
Fuente: Álvarez R., Shagñay S.



Fuente: Álvarez R., Shagñay S.



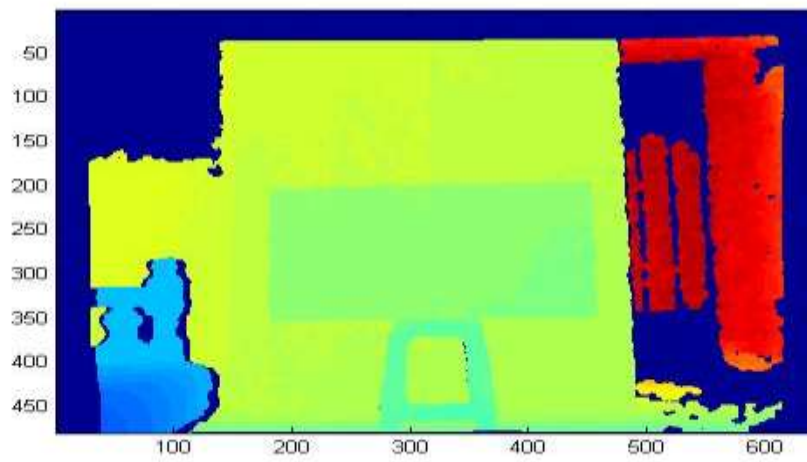
Fuente: Álvarez R., Shagñay S.



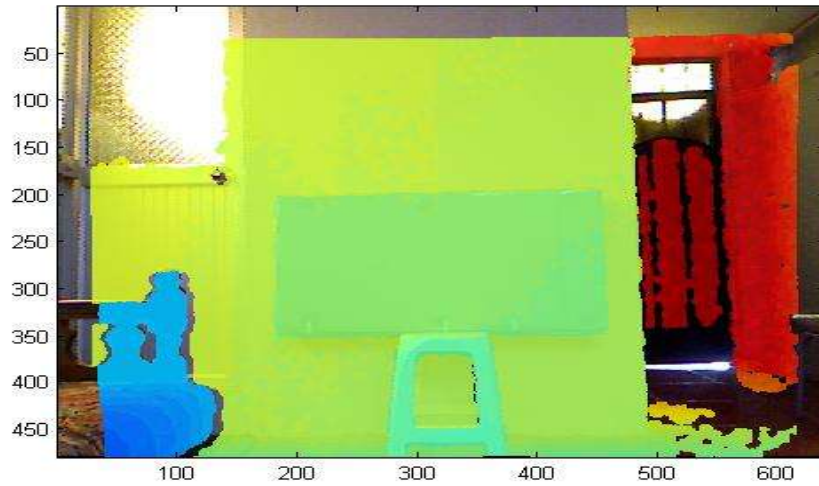
Fuente: Álvarez R., Shagñay S.



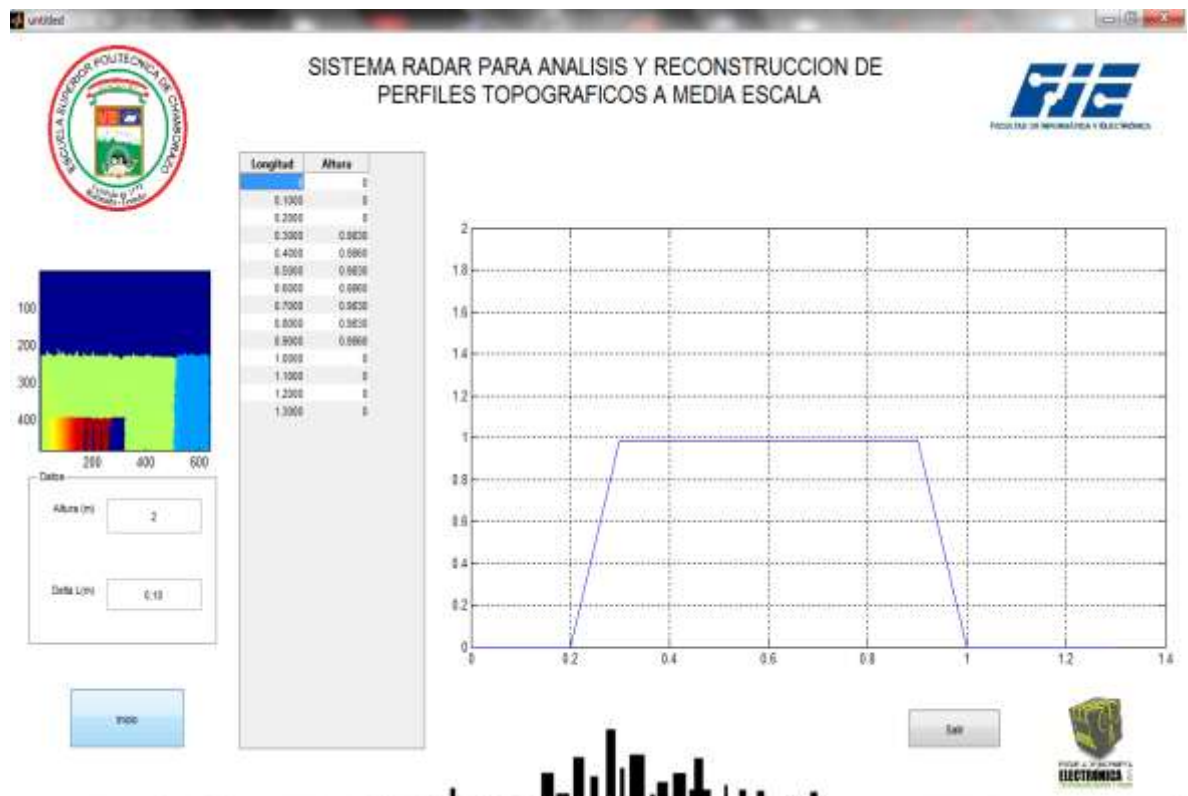
Fuente: Álvarez R., Shagñay S.



Fuente: Álvarez R., Shagñay S.



Fuente: Álvarez R., Shagñay S.



Fuente: Álvarez R., Shagñay S.