

**“DISEÑO DE UN SISTEMA PARA CAPTURA DE GEOMETRÍAS 3D CON
TÉCNICAS DE VISIÓN Y LUZ ESTRUCTURADA”**

**ANDRES ALEJANDRO DIAZ TORO
HERMAN CASTILLO RENDON**

**UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
SAN JUAN DE PASTO
2009**

**“DISEÑO DE UN SISTEMA PARA CAPTURA DE GEOMETRÍAS 3D CON
TÉCNICAS DE VISIÓN Y LUZ ESTRUCTURADA”**

**ANDRES ALEJANDRO DIAZ TORO
HERMAN CASTILLO RENDON**

**Trabajo de grado presentado como requisito parcial para optar al título de
Ingeniero Electrónico**

**Director
Mg. DARIO FERNANDO FAJARDO FAJARDO**

**UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
SAN JUAN DE PASTO
2009**

**“LAS IDEAS Y CONCLUSIONES APORTADAS EN EL TRABAJO DE GRADO,
SON DE RESPONSABILIDAD EXCLUSIVA DE LOS AUTORES”**

**ARTICULO 1 DEL ACUERDO No. 324 DE OCTUBRE 11 DE 1966, EMANADO
DEL HONORABLE CONSEJO DIRECTIVO DE LA UNIVERSIDAD DE NARIÑO**

San Juan de Pasto, Agosto de 2009
Nota de aceptación:

Presidente

Jurado

Jurado

RESUMEN

En esta investigación se presenta el diseño e implementación de un sistema de adquisición de geometría de la superficie de objetos basado en técnicas de visión y luz estructurada, conformado por una cámara, un generador láser y un mecanismo de posicionamiento angular, que permite obtener un modelo tridimensional, el cual puede ser visto bajo un ambiente virtual, constituyéndose en una herramienta útil aplicada en la ingeniería inversa y control de calidad en el área de manufactura de productos.

Este trabajo describe la estructura y elementos de escaneo, el sistema electrónico de control, el software utilizado, el pre-procesamiento y procesamiento digital de imágenes, la calibración del sistema de medidas, la validación de las medidas, recomendaciones para futuras investigaciones, resultados y conclusiones que ésta investigación arrojó.

ABSTRACT

This study presents the design and implementation of an acquisition system of the surface geometry of objects based on vision techniques and structured light, composed by a camera, a laser generator and an angular positioning mechanism that allows to obtain a three-dimensional model, which can be seen under a virtual environment making it a useful tool applied in reverse engineering and quality control in the manufacturing area.

This work describes the structure and elements of scanning, the electronic control system, the software used, the digital images preprocessing and processing, calibration of the measurement system, recommendations for future research, results and conclusions that the research showed.

CONTENIDO

	pág.
INTRODUCCIÓN	18
1. PROBLEMA	19
1.1 DESCRIPCIÓN DEL PROBLEMA	19
1.2 FORMULACIÓN DEL PROBLEMA	19
2. ALCANCE Y DELIMITACIONES	20
3. OBJETIVOS	21
3.1 OBJETIVO GENERAL	21
3.2 OBJETIVOS ESPECIFICOS	21
4. MARCO CONCEPTUAL	22
4.1 DIGITALIZACIÓN TRIDIMENSIONAL	22
4.2 CLASIFICACIÓN DE SISTEMAS DE DIGITALIZACIÓN 3D	22
4.2.1 Digitalización por contacto	23
4.2.2 Digitalización sin contacto	23
4.3 TÉCNICA DE ADQUISICIÓN DE DATOS	24
4.4 VISIÓN POR COMPUTADOR	24
4.5 MORFOLOGÍA MATEMÁTICA EN IMÁGENES BINARIAS	25
4.5.1 Conjuntos	25
4.5.2 Elemento estructurante	27
4.5.3 Operadores morfológicos	27
4.6 ALGORITMOS MORFOLÓGICOS	33

4.6.1 Esqueletización	33
4.6.2 Poda	34
4.7 INGENIERÍA INVERSA	35
4.8 INSPECCION INDUSTRIAL	35
4.9 PRINCIPIO DE TRIANGULACION	36
5. ELEMENTOS DE ILUMINACIÓN Y CAPTURA	38
5.1 LUZ ESTRUCTURADA	38
5.1.1 Láser de línea	39
5.1.2 Seguridad visual	40
5.1.3 Distribución de intensidad	40
5.2 LUZ BLANCA	41
5.2.1 Disposición de los leds de luz blanca	41
5.3 CÁMARA	42
5.3.1 Disposición de la cámara y el láser	43
6. SISTEMA ELECTRÓNICO	45
6.1 MODULO DE CONTROL	45
6.2 MODULO DE DESPLAZAMIENTO	46
6.3 MODULO DE SENSADO	46
6.4 CIRCUITO ELECTRÓNICO	47
7. ESCANEEO DEL OBJETO	48
8. PREPROCESAMIENTO DE IMÁGENES	50
8.1 SUSTRACCION DE IMÁGENES	50
8.2 CONVERSIÓN DE RGB A ESCALA DE GRISES	51

8.3 TRANSFORMACIÓN DE INTENSIDAD	52
8.4 TRANSFORMACIÓN DE UMBRAL	55
8.4.1 Determinación del Umbral	55
8.5 IMAGEN BINARIA	57
9. PROCESAMIENTO DE IMÁGENES	59
10. CALIBRACIÓN	62
10.1 CURVAS DE CONVERSIÓN	65
10.1.1 Curva de conversión en U	65
10.1.2 Curva de conversión en V	68
11. COLOR	75
12. CARACTERIZACIÓN DEL SISTEMA	77
12.1 RESOLUCIÓN	77
12.1.1 Resolución en profundidad ρ	77
12.1.2 Resolución en altura Z	77
12.2 PRECISIÓN EN EL DESPLAZAMIENTO ANGULAR	78
12.3 RANGO	78
12.4 DENSIDAD DE LA NUBE DE PUNTOS	79
12.5 CARACTERÍSTICAS ELÉCTRICAS	80
13. VALIDACIÓN	81
13.1 CILINDRO	81
13.2 PRISMA RECTANGULAR	85
13.3 ESFERA	87
13.4 CÁLCULO DEL ERROR	90

14 RESULTADOS	92
14.1 SOFTWARE DEL DIGITALIZADOR	102
14.1.1 Software de captura	102
14.1.2 Software de calibración	105
14.1.3 Software de análisis	106
14.2 DIGITALIZADOR Y SUS PARTES FISICAS	106
15. CONCLUSIONES	110
16. RECOMENDACIONES	112
BIBLIOGRAFIA	113
ANEXOS	114

LISTA DE TABLAS

	pág.
Tabla 1 Operaciones entre conjuntos	27
Tabla 2 Operadores morfológicos	28
Tabla 3 Especificaciones del láser	39
Tabla 4 Especificaciones de la cámara	42
Tabla 5 Densidad de la nube de puntos y tiempo de escaneo	80
Tabla 6 Consumo de corriente	80

LISTA DE FIGURAS

		pág.
Figura 1	Operaciones sobre los conjuntos A y B	26
Figura 2	Operación morfológica de dilatación	28
Figura 3	Operación morfológica de erosión	29
Figura 4	Operación morfológica de apertura	30
Figura 5	Interpretación geométrica de la apertura	31
Figura 6	Operación morfológica de cierre	31
Figura 7	Interpretación geométrica del cierre	32
Figura 8	Operaciones morfológicas	33
Figura 9	Proceso morfológico de esqueletización	33
Figura 10	Obtención del esqueleto en una imagen A	34
Figura 11	Esqueletización	34
Figura 12	Proceso de poda	34
Figura 13	Principio de la triangulación espacial	36
Figura 14	Dimensiones del láser	39
Figura 15	Angulo de abertura y distribución de intensidad	40
Figura 16	Ubicación de los leds de luz blanca	41
Figura 17	Ubicación de la cámara y el láser	43
Figura 18	Estructura mecánica	43
Figura 19	Eslabón, cámara y láser	44
Figura 20	Diagrama de bloques del sistema electrónico	45
Figura 21	Circuito electrónico	47
Figura 22	Escaneo del objeto	48
Figura 23	Vista superior del proceso de escaneo	49
Figura 24	Línea láser deformada por el objeto	50
Figura 25	Imagen del objeto sin proyección de la línea láser	50
Figura 26	Resultado de la sustracción de imágenes	51
Figura 27	Imagen en RGB	51
Figura 28	Imagen en escala de grises	51
Figura 29	Transformación de intensidad utilizando imadjust	53
Figura 30	Imagen inicial	53
Figura 31	Resultado de la transformación de intensidad	53
Figura 32	Histograma de la imagen inicial	54
Figura 33	Histograma de la imagen con transf. de intensidad	54
Figura 34	Transformación de Umbral	55
Figura 35	Histograma característico de las imágenes	56
Figura 36	Umbral ubicado un 10% antes del pico máximo	56
Figura 37	Ejemplo de imágenes binarias	57
Figura 38	Conversión a imagen binaria	58

Figura 39	Imagen binaria inicial	59
Figura 40	Imagen con apertura	60
Figura 41	Imagen con cierre	60
Figura 42	Imagen esqueletizada	61
Figura 43	Imagen con poda	61
Figura 44	Plano con franjas horizontales	62
Figura 45	Plano ubicado sobre el marco indicando la dirección U y V	62
Figura 46	Marco ubicado sobre un diámetro de la base	63
Figura 47	Disposición marco, cámara y láser	63
Figura 48	Desplazamiento radial de la cámara y el láser	63
Figura 49	Línea láser y esqueletizada	64
Figura 50	Gráfica del desplazamiento radial en milímetros y desplazamiento en píxeles con respecto a la línea de referencia y en dirección U	66
Figura 51	Superposición de las imágenes con líneas esqueletizadas	67
Figura 52	Vista superior del corte transversal de un objeto, cámara y láser	67
Figura 53	Descripción gráfica del radio y del desplazamiento en píxeles en dirección U	68
Figura 54	Imagen en escala de grises	69
Figura 55	Imagen binaria con reducción de espesor en la intersección con las líneas negras	69
Figura 56	Zonas de intersección	70
Figura 57	Centro geométrico de cada zona y errores como puntos muy cercanos y en los extremos	70
Figura 58	Corrección de intersección	70
Figura 59	Gráfica del factor de conversión y el desplazamiento radial	71
Figura 60	Superposición de las imágenes con puntos de intersección	72
Figura 61	Coordenadas cilíndricas	72
Figura 62	Obtención de coordenadas cilíndricas y cartesianas	73
Figura 63	Convención de coordenadas de las imágenes	73
Figura 64	Perfil de una pieza cerámica	75
Figura 65	Perfil	76
Figura 66	Modelo	76
Figura 67	Rango de medidas	79
Figura 68	Objeto a digitalizar	81
Figura 69	Modelo tridimensional	81
Figura 70	Corte transversal del objeto digitalizado	82
Figura 71	Círculo ideal	82
Figura 72	Comparación	82
Figura 73	Círculo ideal y escaneado concéntricos	83
Figura 74	Cilindro digitalizado con centro en (26,-4)	83
Figura 75	Corte transversal	84
Figura 76	Comparación	84

Figura 77	Circulo ideal y escaneado concéntricos	84
Figura 78	Objeto a digitalizar	85
Figura 79	Modelo tridimensional	85
Figura 80	Corte transversal	86
Figura 81	Rectángulo ideal	86
Figura 82	Comparación	86
Figura 83	Datos rotados -11°	87
Figura 84	Comparación	87
Figura 85	Objeto a digitalizar	88
Figura 86	Modelo tridimensional	88
Figura 87	Corte longitudinal	88
Figura 88	Comparación	89
Figura 89	Datos con centro en el origen	90
Figura 90	Error del instrumento de medición	91
Figura 91	Fotografía de un Santa Claus	92
Figura 92	Nube de puntos en Matlab	92
Figura 93	Modelo con información cromática en Matlab	93
Figura 94	Estructura alámbrica y superficie (renderizada) en Rhinoceros	93
Figura 95	Fotografía de un cráneo humano	94
Figura 96	Nube de puntos en Matlab	94
Figura 97	Modelo con información cromática en Matlab	95
Figura 98	Estructura alámbrica y superficie (renderizada) en Rhinoceros	95
Figura 99	Fotografía de Garfield	96
Figura 100	Nube de puntos en Matlab	96
Figura 101	Modelo con información cromática en Matlab	97
Figura 102	Estructura alámbrica y superficie (renderizada) en Rhinoceros	97
Figura 103	Fotografía de un faraón	98
Figura 104	Nube de puntos en Matlab	98
Figura 105	Modelo con información cromática en Matlab	99
Figura 106	Estructura alámbrica y superficie (renderizada) en Rhinoceros	99
Figura 107	Fotografía de un bebé	100
Figura 108	Nube de puntos en Matlab	100
Figura 109	Modelo con información cromática en Matlab	101
Figura 110	Estructura alámbrica y superficie (renderizada) en Rhinoceros	101
Figura 111	Software de captura	102
Figura 112	Etapa de pre-visualización	103
Figura 113	Software de calibración	105
Figura 114	Software de análisis	106
Figura 115	Digitalizador	106

Figura 116	Digitalizador y puerta deslizable	107
Figura 117	Parte interior del digitalizador	107
Figura 118	Placa de control	108
Figura 119	Eslabón de de desplazamiento radial y angular	108
Figura 120	Cámara y Láser	109
Figura 121	Placa del circuito de control.	114

LISTA DE ANEXOS

	pág.
Anexo A. Placa del circuito de control.	114
Anexo B. Diagrama de flujo principal.	115
Anexo C. Diagrama de flujo de la función Digitalizar.	116
Anexo D. Código fuente del programa de digitalización.	118

GLOSARIO

ALGORITMO: conjunto ordenado y finito de operaciones que permite hallar la solución de un problema.

DRIVER: circuito integrado que permite entregar energía a un motor paso a paso, de acuerdo al estado de sus entradas.

ESLABÓN: estructura que hace parte del brazo de un robot.

ESTEREOSCÓPICO: aparato con capacidad de fundir dos imágenes para dar la sensación de relieve.

FOTÓN: partícula elemental de masa nula, que representa la unidad cuántica de radiación electromagnética según la teoría de M. Planck.

HISTOGRAMA: representación gráfica de una distribución de frecuencias por medio de rectángulos, cuyas anchuras representan intervalos de la clasificación y cuyas alturas representan las correspondientes frecuencias.

IDEMPOTENTE: dicho de un elemento de un conjunto: Que tiene la propiedad de que al multiplicarse por sí mismo vuelve a obtenerse el mismo elemento.

IMAGEN RGB: imagen que muestra al objeto con sus colores reales por medio de la combinación de tonos rojos, verdes y azules.

LÓBULO: saliente de un borde en forma de onda.

LUZ ESTRUCTURADA: radiación electromagnética que está enfocada a ciertos lugares generando determinadas formas o figuras.

MONOCULAR: la imagen obtenida solo tiene un sensor de origen.

MOTOR PASO A PASO: motor que se mueve por medio de secuencias digitales.

PÍXEL: la parte más pequeña de una imagen.

TOOLBOX (CAJA DE HERRAMIENTAS): los diferentes comandos que contiene un software para realizar tareas de cierto tipo.

TRIANGULACIÓN: disponer las piezas de una armazón, de modo que formen triángulo.

INTRODUCCION

En los últimos años, los sistemas de adquisición de datos por computador han tomado especial importancia debido a los avances que se han producido desde el punto de vista computacional y gracias a la integración y desarrollo de nuevos sensores. Fruto de estos avances tecnológicos se ha producido, primero, el desarrollo de escáneres 2D que permiten digitalizar objetos planos y después, la aparición de escáneres 3D que permiten digitalizar objetos con volumen.

La implementación de los sistemas de digitalización de piezas industriales o artísticas tiene dos posibles finalidades. La primera es, la etapa de inspección del producto, donde se analizan las desviaciones entre el modelo CAD estándar y los datos de los puntos medidos en la pieza. Se recogen grandes cantidades de puntos y el análisis de esa información permite mejorar la calidad de la pieza. La segunda finalidad de la digitalización es crear un modelo CAD, capturando la forma de una pieza real, que es lo que se conoce como ingeniería inversa. Esto es especialmente útil en objetos complejos en términos de sus formas, contornos y perfiles, elementos clave en el modelado de productos. Tal es el caso del diseño de piezas mecánicas, moldes, objetos ergonómicos o estéticos, gráficos para la industria del entretenimiento y formas aerodinámicas.

El sistema propuesto está encaminado a adquirir información tridimensional de una pieza, basado en un método óptico, específicamente el principio de triangulación activa por proyección de luz estructurada, escogido por su versatilidad de escaneo y exactitud. Este método se fundamenta en la proyección sobre las piezas de un patrón de iluminación, que en éste caso corresponde a un plano láser, de forma que se tienen coordenadas tridimensionales de los puntos de la superficie de la pieza a partir de la imagen proporcionada por un sensor de video mediante triangulación.

1. PROBLEMA

1.1 DESCRIPCIÓN DEL PROBLEMA

En Colombia la visión por computador es un campo que se ha estado estudiando desde hace diez años aproximadamente y por lo tanto la tecnología que se deriva de su investigación no es fácil de conseguir en el mercado nacional, resultando desconocida para las empresas que de alguna manera se verían beneficiadas utilizándola. Esta falta de información disminuye la posibilidad de abrir el camino a nuevas vías de desarrollo y limita el avance de las investigaciones en el tema.

Además estos sistemas comercialmente son muy costosos resultando difíciles de adquirir; la mayoría de proveedores se encuentran en países lejanos, lo cual dificulta su envío y aumenta considerablemente su precio.

Existen dos métodos de visión por computador, el activo y el pasivo. En este proyecto se utilizara el método de visión activo debido a que utiliza una fuente de luz estructurada, que permite obtener mejores resultados que el método pasivo de visión por computador.

1.2 FORMULACIÓN DEL PROBLEMA

¿Cómo capturar la geometría de un objeto para generar un modelo tridimensional que lo represente adecuadamente?

2. ALCANCE Y DELIMITACIONES

El alcance de este proyecto es el diseño y construcción de un instrumento que genere coordenadas tridimensionales a partir de la proyección de una línea láser sobre un objeto sólido a ser digitalizado y el procesamiento de las imágenes obtenidas durante el escaneo, dando como resultado un modelo tridimensional que brinde la posibilidad de ser visto desde diferentes perspectivas bajo un ambiente virtual.

Además está diseñado para objetos sólidos y con las siguientes restricciones:

- Objetos no mayores a 20cm de alto.
- Objetos no menores a 3cm de alto y de diámetro.
- Objetos que alcancen a ser detectados por la cámara.
- Objetos con propiedades no reflejantes
- Objetos opacos

Para una mejor visualización de los modelos tridimensionales se les adicionó información cromática, sin ser éste un objetivo del proyecto.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Escanear y modelar en forma gráfica tridimensional un objeto utilizando luz estructurada

3.2 OBJETIVOS ESPECÍFICOS

- Recolectar y analizar información acerca de la técnica de visión y luz estructurada
- Diseñar y construir la estructura mecánica y electrónica que permita posicionar el módulo cámara-láser para escanear el objeto.
- Calibrar los sensores de imágenes para obtener datos con alta precisión y exactitud.
- Procesar las imágenes y obtener los patrones proyectados en el objeto utilizando Matlab.
- Reconstruir y visualizar el resultado del procesamiento de los datos a través de una gráfica tridimensional.
- Validar los datos obtenidos.

4. MARCO CONCEPTUAL

4.1 DIGITALIZACIÓN TRIDIMENSIONAL

“El término ‘Digitalización 3D’, apareció a finales de la década de los 90, designa las técnicas de adquisición de medidas tridimensionales. Los scanners 3D son dispositivos que nos permiten construir modelos CAD (Diseño Asistido por Computador) a partir de objetos físicos, ya sean partes del cuerpo humano o piezas industriales”¹.

Estos sistemas de adquisición de la geometría y el color de objetos basados en la Visión por Computador se han desarrollado rápidamente debido a la gran variedad de aplicaciones en las que estas informaciones son necesarias. Las nuevas técnicas de digitalización 3D nos permiten capturar los puntos de medida con mayor precisión y velocidad, pudiéndose emplear sobre distintos tipos de objetos, de distintas dimensiones, geometrías y texturas.

“Las mejoras en la tecnología de medición óptica han hecho posible capturar geometrías tridimensionales con relativa facilidad. Son muchos los sistemas de reconstrucción de geometrías que han sido desarrollados; sin embargo, no ha sido establecida una metodología general y automatizada para crear modelos geométricos a partir de dicha tecnología”². El diseño de un sistema como éste, depende del tamaño, forma, ubicación y naturaleza de los objetos que se pretende digitalizar.

El aspecto fundamental de la implementación de los sistemas de digitalización se encuentra en la posibilidad de reproducir las geometrías de objetos existentes. Esto es especialmente útil en objetos complejos en términos de sus formas, contornos y perfiles. La creciente implementación de la digitalización de productos en una gran diversidad de industrias ha conducido al desarrollo de diferentes sistemas.

4.2 CLASIFICACIÓN DE SISTEMAS DE DIGITALIZACIÓN 3D

“Existen distintos sistemas de digitalización 3D, que principalmente se pueden dividir en dos grandes grupos: sistemas con contacto o sin contacto con el objeto a digitalizar.

¹ LEÓN, Miguel Ángel. www.interempresas.net/MetalMecanica/Articulos/Articulo.asp?A=8697.

² LÓPEZ GUERRERO, F. Eugenio. *Diseño de un instrumento para captura de geometrías 3D con aplicación en la manufactura de productos*. Pág. 2.

4.2.1 Digitalización por contacto. Estos sistemas de digitalización 3D son los más antiguos. Principalmente se emplean para la verificación dimensional de piezas industriales (control de calidad). Con estos sistemas se obtienen las coordenadas de los puntos gracias al desplazamiento de una punta sobre la superficie a digitalizar.

En la actualidad existen cabezales de digitalización en continuo. La velocidad de adquisición de datos es muy alta, ya que éstos no se separan de la superficie a digitalizar.

Otro tipo de sistema de digitalización 3D con contacto son los brazos articulados de operación manual. Estos poseen una elevada precisión; pero por el contrario tienen una velocidad de adquisición de datos muy baja, ya que se necesita llevar manualmente la punta a cada punto que se quiera digitalizar, con lo cual la digitalización de un objeto de tamaño medio resulta muy tediosa.

Para emplear estos sistemas por contacto, se necesita que las piezas tengan la rigidez suficiente para que no se deformen con contacto de la punta y debido a la geometría de las puntas, es imposible digitalizar algunas ranuras y ángulos interiores; pero poseen una elevada resolución.

4.2.2 Digitalización sin contacto. La principal ventaja de los digitalizadores 3D sin contacto es que tienen una velocidad de adquisición de datos muy superior a las de los digitalizadores con contacto. Podemos dividir las técnicas de digitalización sin contacto en dos grandes grupos:

* Técnicas de visión pasiva. El sistema visual humano permite obtener información de profundidad mediante la fusión de dos escenas monoculares, que son las escenas que captan cada uno de nuestros ojos. Este sistema nos permite "ver en 3D" y es conocido como principio de visión estereoscópica. Así pues, dicho principio se basa en utilizar dos puntos de vista de un mismo objeto para encontrar las coordenadas tridimensionales. Para determinar la posición de un punto a partir de dos imágenes es necesario tener un modelo del sistema óptico utilizado.

Este principio general puede mejorarse con modelos de cámaras más elaborados o utilizando más de dos cámaras, lo que se conoce con el nombre de fotogrametría. La visión estereoscópica presenta el interés de poder dar la posición en 3D de puntos sobre las superficies independientemente de la iluminación específica (visión pasiva).

* Técnicas de visión activa. Estas técnicas son las que hacen intervenir una fuente de luz específica para determinar las coordenadas tridimensionales de los puntos de medida. Los sistemas ópticos se fundamentan en el cálculo de la profundidad.

Estos sistemas constan siempre como mínimo de un emisor de luz y un receptor. Conociendo la dirección del rayo emitido y la del recibido se obtienen las dimensiones del triángulo y por lo tanto la profundidad del punto inspeccionado”³.

4.3 TÉCNICA DE ADQUISICIÓN DE DATOS

“Los dispositivos ópticos activos son basados en un emisor, que produce algún tipo de iluminación estructurada sobre un objeto a recuperarse, y un sensor que es típicamente una cámara CCD, la cual adquiere la imagen del patrón distorsionado reflejado por la superficie del objeto”⁴.

Cuando se habla de la obtención de la información de profundidad también se habla de triangulación, porque permite conocer algunas variables como las posiciones relativas entre sensor y emisor. El emisor puede producir distintas formas de luz estructurada como puntos, franjas o patrones más complejos y éstas pueden ser proyectadas por medio de láseres, proyectores de luz blanca, proyectores de diapositivas y proyectores de video digitales.

4.4 VISIÓN POR COMPUTADOR

La “*Visión por Computador*” se refiere a las técnicas y modelos desarrollados con el fin de procesar, analizar y explicar información específica obtenida a través de imágenes digitales.

Desde sus inicios la Visión por Computador ha inspirado sus desarrollos en el estudio del sistema visual humano el cual sugiere la existencia de diferentes tipos de tratamiento de la información visual dependiendo de metas u objetivos específicos, es decir, la información visual percibida es procesada en distintas formas con base en las características particulares de la tarea a realizar, por lo que la Visión por Computador propone varias técnicas que permiten obtener una representación del mundo a partir del análisis de imágenes obtenidas desde cámaras de video”⁵.

³ LEÓN, Miguel Ángel. www.interempresas.net/MetalMecanica/Articulos/Articulo.asp?A=8697.

⁴ SOANCATL AGUILAR, Venustiano. *Recuperación de Información tridimensional usando luz estructurada*. Pág. 7.

⁵ iabot.iespana.es/ciencia/software/ia/reconocimiento_de_imagenes.htm

La visión para el ser humano es una de sus mayores fuentes de información, por lo tanto se cree conveniente implementarla en computadoras para que puedan usarla en conjunto con mecanismos de aprendizaje y así detectar y ubicar objetos en el mundo real.

4.5 MORFOLOGÍA MATEMÁTICA EN IMÁGENES BINARIAS

La morfología matemática es una herramienta muy utilizada en el procesamiento digital de imágenes. Las operaciones morfológicas pueden simplificar los datos de una imagen, preservar las características esenciales y eliminar aspectos irrelevantes.

Teniendo en cuenta que la identificación y descomposición de objetos, la extracción de rasgos, la localización de defectos como los encontrados en líneas de ensamblaje, están sumamente relacionados con las formas, la morfología matemática desempeña un papel importante en estos aspectos.

“La morfología matemática se puede usar, entre otros, con los siguientes objetivos:

- Pre-procesamiento de imágenes (supresión de ruido).
- Destacar la estructura de objetos (extraer el esqueleto, marcado de objetos, envolvente convexa, ampliación, reducción).
- Descripción cualitativa de objetos (área, perímetro, diámetro, etc)”⁶.

“Los tres elementos necesarios para realizar un proceso morfológico son:

- Conjuntos (Imágenes)
- Elementos Estructurantes (EE)

Operadores Morfológicos (dilatación, erosión, apertura/cierre)”⁷.

4.5.1 Conjuntos. El lenguaje que se usa en las técnicas morfológicas proviene de la teoría de conjuntos. Cada conjunto representa la forma de los objetos en una imagen.

Una imagen binaria está descrita completamente por dos conjuntos, uno compuesto por todos los píxeles negros y otro por todos los píxeles blancos, donde:

⁶ Operaciones morfológicas básicas. *Cap5.pdf*

⁷ www.tsc.uc3m.es/imagene/Curso_ProcesadoMorfologico/index.html

- Blanco=1
- Negro=0

Cada conjunto define un plano así:

$$\text{Primer plano: } \mathbf{A} = \{(x,y) | f(x,y) = 1\} \quad (1)$$

$$\text{Segundo plano o Fondo: } \mathbf{B} = \{(x,y) | f(x,y) = 0\} \quad (2)$$

En una imagen binaria, los conjuntos existentes son puntos de un espacio 2D: cada elemento es un punto de coordenadas (x, y) en el plano bidimensional.

En virtud de la equivalencia entre imágenes binarias y conjuntos, cualquier transformación sobre un conjunto define una nueva imagen binaria.

* Operaciones sobre conjuntos.

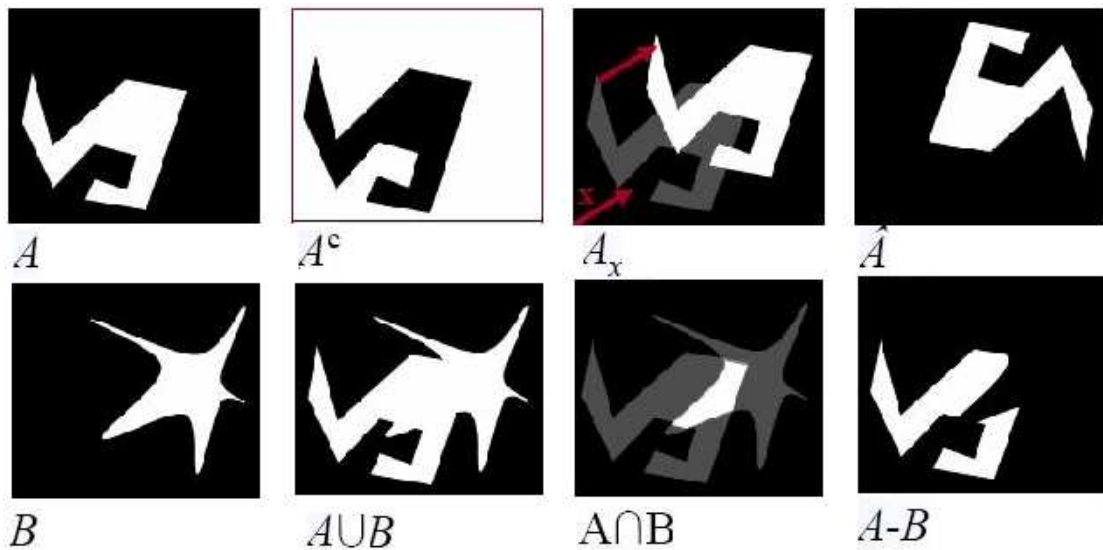


Fig. 1, Operaciones sobre los conjuntos A y B

Sean A y B conjuntos de puntos en un espacio Z^2 , como se muestra en la figura 1, con componentes $\mathbf{a} = (a_1, a_2)$, $\mathbf{b} = (b_1, b_2)$, se presentan las siguientes operaciones:

Complemento $A_c = A^c = \{(x,y) | (x,y) \notin A\}$

Traslación: $A_x = A + \mathbf{x} = \{\mathbf{a} + \mathbf{x} | \mathbf{a} \in A\}$ Con $\mathbf{X} = (x, y)$

Reflexión: $\hat{A} = -A = \{-\mathbf{a} \mid \mathbf{a} \in A\} \quad -\mathbf{a} = (-a_1, -a_2)$

Diferenciación: $A-B = \{\mathbf{a} \mid \mathbf{a} \in A, \mathbf{a} \notin B\} = A \cap B^c$

Los operadores morfológicos transforman conjuntos en conjuntos. En virtud de la correspondencia entre conjuntos e imágenes binarias, por cada operador morfológico puede construirse una transformación equivalente entre imágenes binarias como indica la tabla 1.

→	$c(x,y) = a(x,y) \text{ AND } b(x,y)$	↔	$C = A \cap B$
→	$c(x,y) = a(x,y) \text{ OR } b(x,y)$	↔	$C = A \cup B$
→	$c(x,y) = \text{NOT } a(x,y)$	↔	$C = A^c$
→	$c(x,y) = a(x,y) \text{ XOR } b(x,y)$	↔	$C = (A \cap B^c) \cup (A^c \cap B)$
→	$c(x,y) = a(x-x_0, y-y_0)$	↔	$C = A + (x_0, y_0)$
→	$c(x,y) = a(-x, -y)$	↔	$C = \hat{A}$

Tabla 1, Operaciones entre conjuntos

Estas equivalencias son la base de la implementación de los algoritmos morfológicos en procesamiento de imágenes.

4.5.2 Elemento Estructurante. El elemento estructurante *EE* es un conjunto de puntos formando una pequeña imagen que determinará la estructura de la imagen sobre la que se aplicará la operación morfológica. Este será el responsable de la forma y el tamaño de los objetos que forman la imagen. Los operadores morfológicos analizan la estructura geométrica de una imagen usando como sonda un patrón de ajuste que se denomina Elemento Estructurante (E.E.).

El operador analiza típicamente su posición en relación al primer plano y al fondo de la misma, desplazando el EE sobre la imagen. Su centro se sitúa en cada píxel de la imagen original, aplicando la operación morfológica sobre los puntos situados bajo el EE.

4.5.3 Operadores Morfológicos.

OPERADOR	SÍMBOLO	TIPO
Dilatación	$\delta_C(A)$	Extensivo
Erosión	$\epsilon_C(A)$	Anti-extensivo

Apertura	$\gamma_B(A)$	Anti-extensivo
Cierre	$\phi_B(A)$	Extensivo

Tabla 2, Operadores Morfológicos

Una operación morfológica es una transformación de un conjunto cuyo resultado es otro conjunto.

La operación morfológica $\Phi(A)$ aplicada a una imagen consiste en desplazar el EE por toda la imagen. Si posicionamos el EE sobre algún punto de la imagen, el píxel de la imagen sobre el que se superpone se denomina píxel actual. El resultado de la relación entre el EE y la imagen en la posición actual se almacena en el píxel actual de la imagen.

Las características más importantes de un operador morfológico son:

- Extensivo: si el resultado contiene a la imagen original.

$$A \subset \Phi(A)$$

- Anti-extensivo: si el resultado está contenido en la imagen original.

$$\Phi(A) \subset A$$

Idempotente: si al aplicar el operador dos o más veces, el resultado es el mismo que si se aplicara una sola vez.

$$\Phi(\Phi(A)) \subset \Phi(A)$$

* Dilatación.

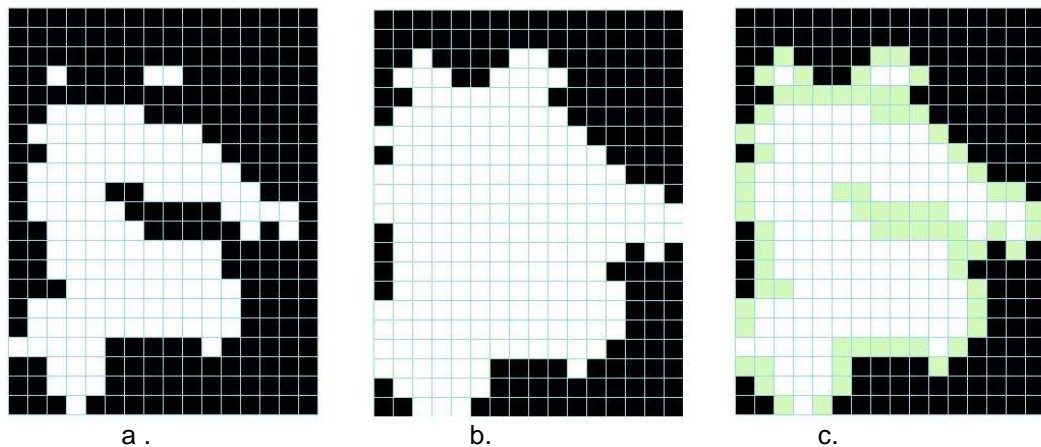
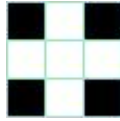


Fig. 2, Operación morfológica de Dilatación;
a. Imagen original A b. Imagen Dilatada c. Píxeles que han cambiado de negro a blanco

Elemento estructurante B:



Dada una imagen A y un elemento estructurante B (figura 2), la dilatación de A por B se define como:

$$\delta_B(A) = A \oplus B = \{x \text{ tal que } (\tilde{B})_x \cap A \neq \phi\} \quad (3)$$

Se obtiene en base a la reflexión de B con respecto a su origen y un desplazamiento x.

La salida de la dilatación es el conjunto de puntos barridos por el centro del EE mientras algún punto de B coincide con alguno de A. Alternativamente, puede interpretarse la dilatación como el resultado de reemplazar cada píxel blanco de la imagen original por una réplica del elemento estructurante. La dilatación es extensiva.

* Erosión.

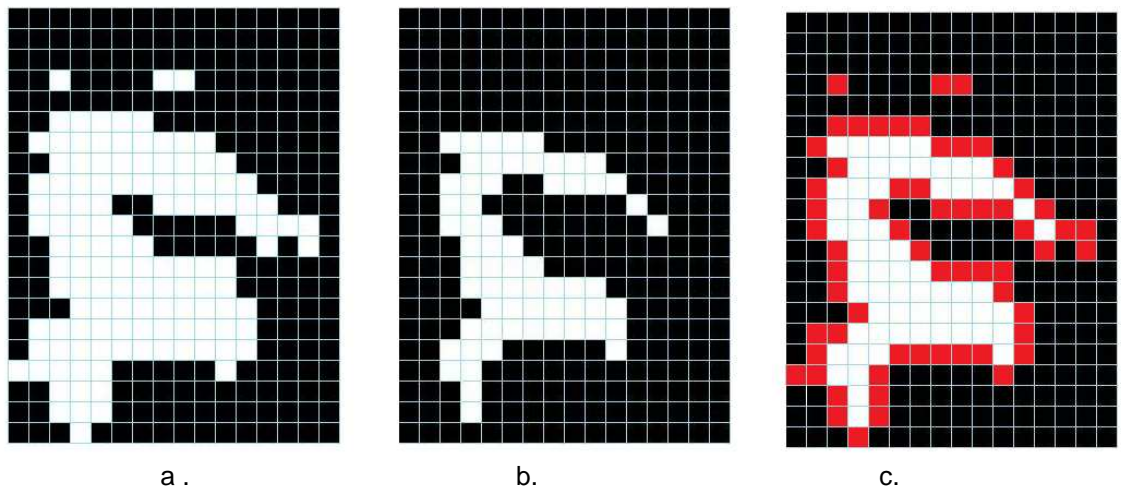
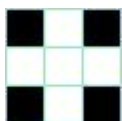


Fig. 3, Operación morfológica de Erosión;
a. Imagen original A b. Imagen erosionada c. Píxeles que han cambiado de blanco a negro

Elemento estructurante B:



Dada una imagen A y un elemento estructurante B (figura 3), la erosión de A por B se define como:

$$\epsilon_B(A) = A \ominus B = \{X \text{ tal que } B_x \subset A\} \quad (4)$$

Es el conjunto de todos los puntos X, tales que B trasladado X está contenido en A.

La salida de la erosión es el conjunto de puntos barridos por el centro del EE mientras se cumpla que todos los puntos de B estén contenidos en A, eliminando grupos de píxeles donde el EE no cabe. La erosión es anti-extensiva.

* Apertura.

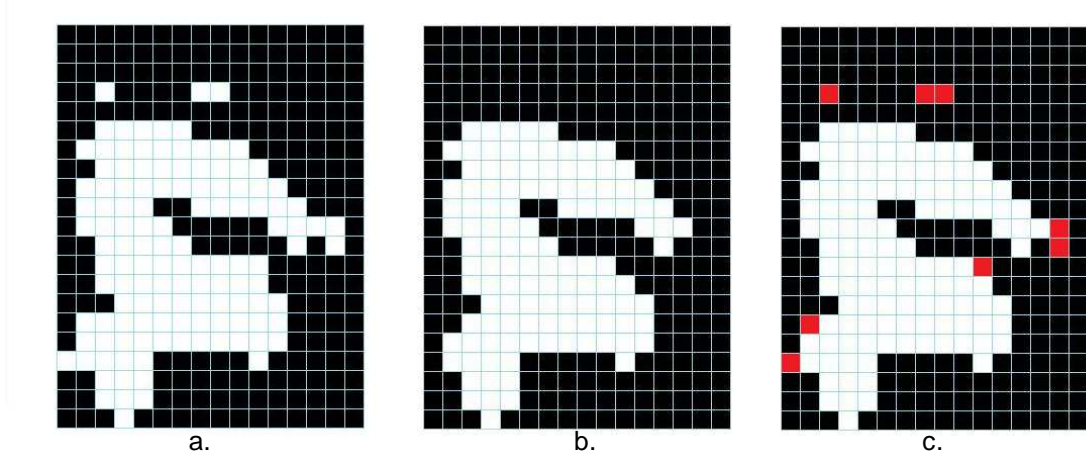
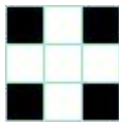


Fig. 4, Operación morfológica de Apertura; a. Imagen original A b. Imagen luego de hacer la apertura c. Píxeles que han cambiado de blanco a negro

Elemento estructurante B:



Dada una imagen A y un elemento estructurante B (figura 4), la apertura de A por B se define como:

$$\gamma_B(A) = A \circ B = (A \ominus B) \oplus B = \delta_B(\epsilon_B(A)) \quad (5)$$

Se observa que la apertura combina el operador de erosión y el operador de dilatación utilizando el mismo elemento estructurante. Se obtiene desplazando el EE B por el interior del conjunto y eliminando las zonas por las que B no puede "pasar".

Las características de la apertura son:

- Es un operador anti-extensivo: $A \circ B$ es un subconjunto de A .
- Es un operador idempotente: $(A \circ B) \circ B = A \circ B$.

La interpretación geométrica de este operador morfológico es:

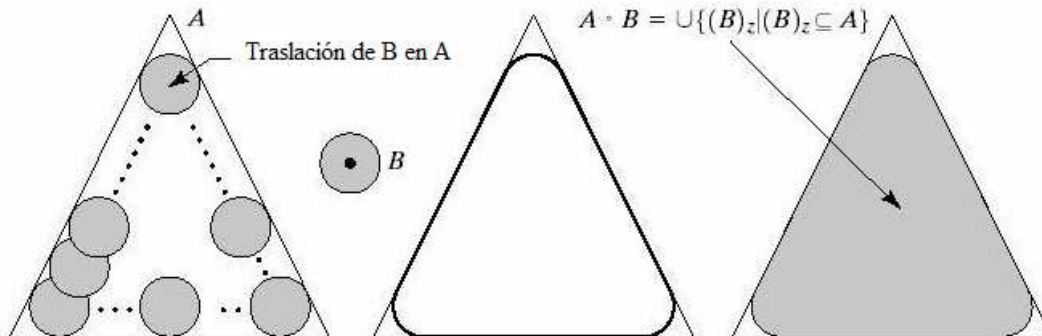


Fig. 5, Interpretación geométrica de la apertura

Los efectos sobre la imagen son:

- Alisa contornos (redondear las esquinas donde no quepa el EE).
- Eliminar las protuberancias donde no quepa el EE.
- Separa objetos en puntos estrechos.

* Cierre.

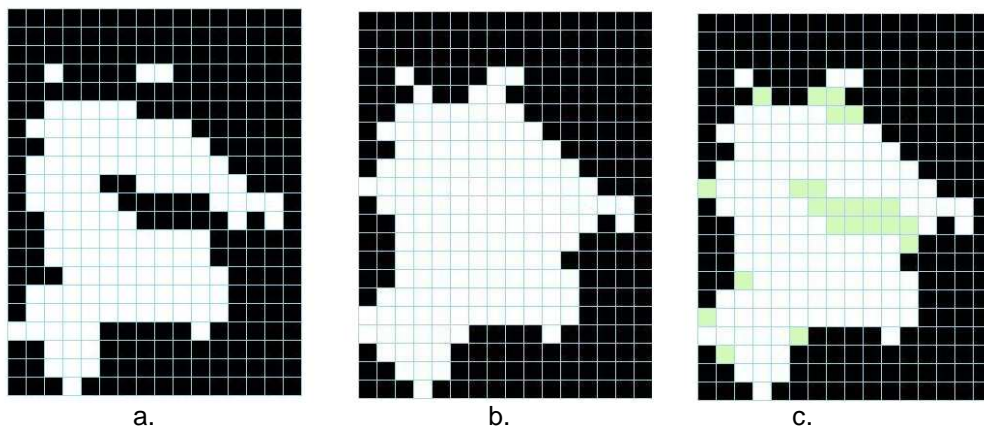
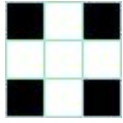


Fig. 6, Operación morfológica de Cierre; a. Imagen original A
b. Imagen luego de hacer el Cierre c. Píxeles que han cambiado de negro a blanco

Elemento estructurante B:



Dada una imagen A y un elemento estructurante B (figura 6), el cierre de A por B se define como:

$$\varphi_B(A) = A \cdot B = A \otimes B = (A \oplus B) \ominus B = \varepsilon_B(\delta_B(A)) \quad (6)$$

El cierre combina el operador de dilatación y el operador de erosión utilizando el mismo elemento estructurante pero en orden contrario al de apertura.

Las características del cierre, son:

- Es un operador extensivo: $A \subset A \cdot B$
- Es un operador idempotente: $(A \cdot B) \cdot B = A \cdot B$

La interpretación geométrica de este operador morfológico es:

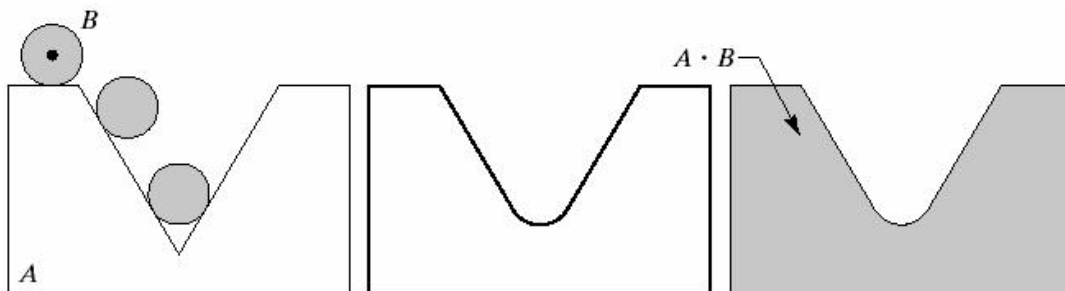


Fig. 7, Interpretación geométrica del cierre

Sus efectos sobre una imagen son:

- Tiende a alisar porciones del contorno, fusiona estrechas grietas, y rellena vacíos en el contorno.
- Rellena agujeros pequeños.
- Elimina los entrantes pequeños.
- Conecta objetos vecinos.

* Comparación entre las diferentes operaciones morfológicas.

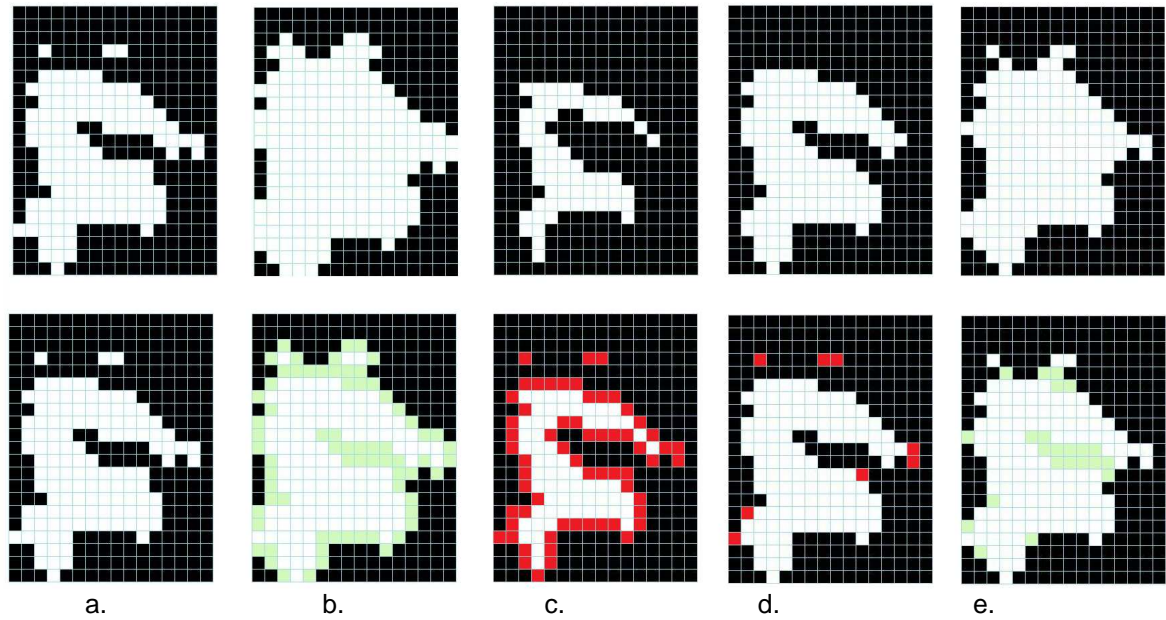


Fig. 8, Operaciones morfológicas; a. Imagen original
b. Dilatación c. Erosión d. Apertura e. Cierre

En la figura 8, se puede comparar el resultado entre las distintas operaciones morfológicas y entender sus diferencias.

4.6 ALGORITMOS MORFOLÓGICOS

4.6.1 Esqueletización.

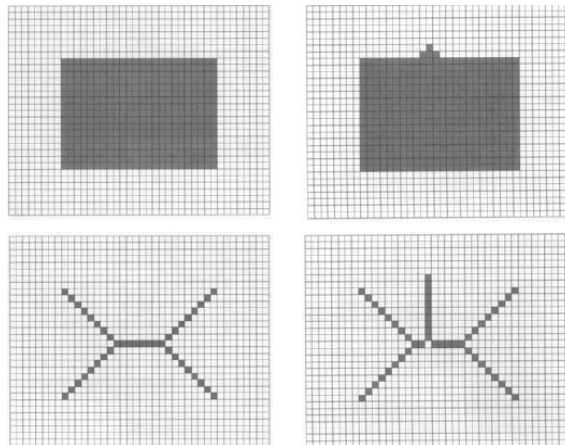


Fig. 9, Proceso Morfológico de esqueletización.

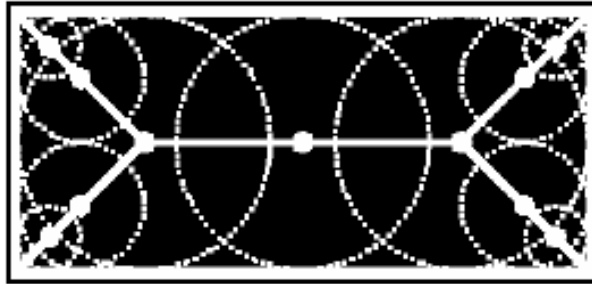


Fig. 10, Obtención del esqueleto en una imagen A

Dada una imagen A , se dice que z pertenece al esqueleto de A si y solo si:

- $(D)z$ es el disco más grande centrado en z y contenido en A , es decir, no existe ningún otro disco más grande conteniendo a $(D)z$ e incluido en A (en tal caso, $(D)z$ se denomina *disco máximo*)
- $(D)z$ toca la frontera de A en, al menos, dos puntos (figura 10).

Su adaptación a imágenes digitales puede ser computacionalmente muy costosa, y suelen utilizarse algoritmos complejos que obtienen buenas aproximaciones con imágenes digitales en un tiempo razonable. Un ejemplo se muestra a continuación:

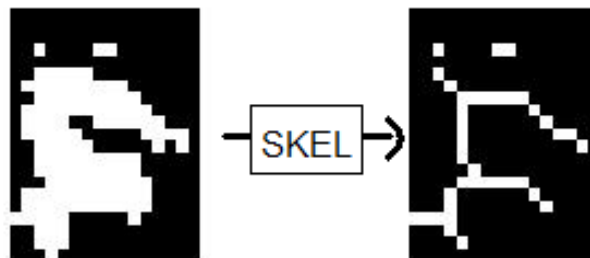


Fig. 11, Esqueletización.

4.6.2 Poda.

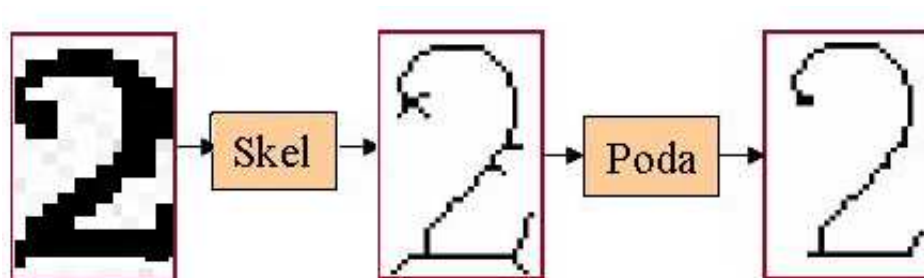


Fig. 12, Proceso de poda.

Este es un paso fundamental a la hora de calcular el esqueleto de una imagen.

Los algoritmos de poda persiguen eliminar componentes no deseados o elementos parásitos resultantes de aplicar operaciones morfológicas de tipo esqueletización como se muestra en la figura 12.

4.7 INGENIERÍA INVERSA

Se conoce como Ingeniería Inversa o *Reverse Engineering* al proceso de duplicar una pieza, componente o conjunto, sin la ayuda de planos, documentación o modelos de CAD auxiliares. Se parte siempre de un modelo físico y se usan métodos de ingeniería de medida, análisis y diseño para finalmente obtener una réplica idéntica o mejora del objeto.

“La Ingeniería Inversa puede ser aplicada en cualquiera de las fases o etapas que componen el proceso de desarrollo de producto, aplicada a cualquier sector industrial (automoción, cosmética, electrónica etc.) así como para cualquier tipología de piezas o producto (piezas estructurales, de diseño, de plástico, metal etc.)”⁸. Los museos virtuales cuentan con modelos obtenidos utilizando ingeniería inversa y permiten una mayor interactividad entre el usuario y el modelo junto con el ambiente virtual en el que se encuentra.

4.8 INSPECCIÓN INDUSTRIAL

La inspección es un proceso de comparación de piezas, objetos individuales o materiales continuos fabricados según una norma, con vistas al mantenimiento de cierto nivel de calidad. Los sistemas de inspección visual automática se basan en la visión artificial para realizar el proceso de inspección.

Las ventajas de la inspección automática son; la independencia de factores humanos, distinción de detalles no visibles por el ojo humano, trabajo por fuera del espectro visible con una velocidad mayor.

“Los sistemas de Inspección Visual Automática se están introduciendo en industrias con procesos productivos que se caracterizan por líneas de producción en serie con múltiples etapas y alto valor añadido de las piezas o alta velocidad de producción”⁹. Al contar con un sistema de inspección automático se mejora la calidad del producto (productos homogéneos) y se aíslan productos que no cumplen con un cierto grado de similitud con el modelo utilizado como referencia.

⁸ www.plastunivers.com/Arees/enginversa/inspeccion.htm

⁹ PÉREZ CORTÉS, Juan Carlos. *Los sistemas de inspección visual automática en los procesos industriales*. Universidad Politécnica de Valencia. Págs. 9, 10, 12, 13.

“La inspección industrial con luz estructurada se da en las líneas de procesamiento de alimentos donde se identifican vegetales con imperfecciones, o también en las líneas de empaquetamiento de fármacos para asegurar que la capsula correcta se encuentra en la botella a la que pertenece”¹⁰. Debido a la variedad de aplicaciones que posee un digitalizador tridimensional se están explorando nuevas técnicas que permitan mejorar su funcionamiento.

4.9 PRINCIPIO DE TRIANGULACIÓN

El principio matemático de triangulación permite la adquisición de información geométrica de la superficie de un objeto cuando se utilizan técnicas de visión activa¹¹. La cámara, el láser y el origen del sistema de referencia conforman los vértices del triángulo como se indica a continuación:

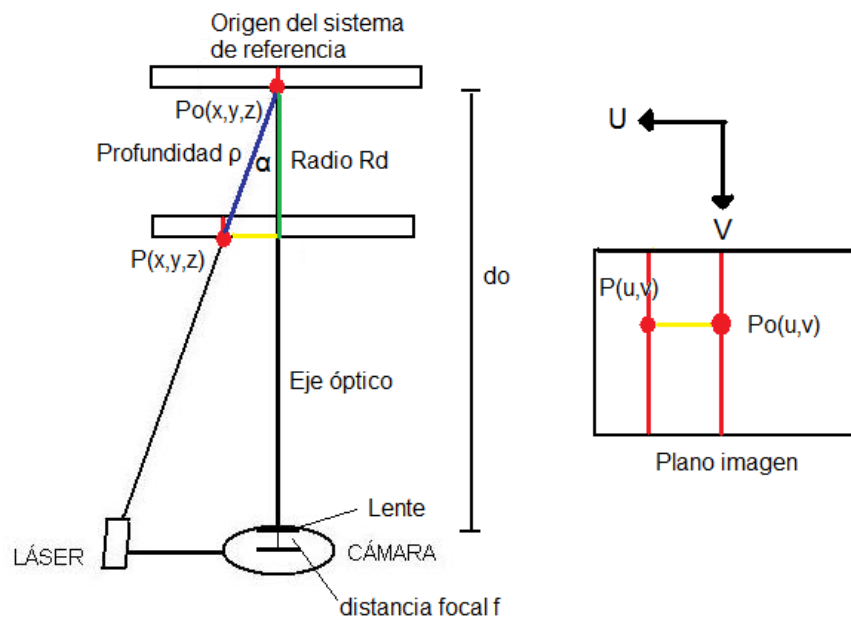


Fig. 13, Principio de triangulación espacial.

De acuerdo con la ley de Gauss para lentes delgadas, la relación que existe entre el radio Rd y la coordenada u del punto P se representa mediante la siguiente ecuación:

$$Rd(u) = \frac{u}{M \cos(\alpha) \left[\tan(\alpha) + \frac{u}{f} \right]} \quad (7)$$

¹⁰ www.stockeryale.com/i/lasers/structured_light.htm

¹¹ PATIÑO VANEGAS, Alberto. *Sistema óptico para la extracción de parámetros metrológicos del pie*. Universidad Industrial de Santander. Págs. 16, 17, 18,19.

Donde M es la razón entre la distancia focal f y la distancia entre el lente y el origen del sistema do así:

$$Rd(U) = \frac{U}{\frac{f}{do} \cos(\alpha) \left[\tan(\alpha) + \frac{U}{f} \right]} \quad (8)$$

La expresión que relaciona la altura Z con las coordenadas u y v del punto P es:

$$z(U, V) = \frac{do}{f} V - \frac{1}{\frac{f}{do} \cos(\alpha) \left[\tan(\alpha) + \frac{U}{f} \right]} UV \quad (9)$$

Estas ecuaciones no se utilizaron porque no se contaba con algunas especificaciones de los instrumentos utilizados como por ejemplo la distancia focal de la cámara. Sin embargo, las ecuaciones se pueden obtener de manera experimental como se indica en el capítulo 10, que corresponde a la calibración del sistema.

5. ELEMENTOS DE ILUMINACIÓN Y CAPTURA

5.1 LUZ ESTRUCTURADA

“Dentro del campo de la Visión tridimensional, existen un gran número de técnicas que hoy son empleadas con éxito en numerosas aplicaciones industriales. Entre todas ellas, se encuentra lo que se conoce como Luz Estructurada.

Este tipo de sistema es un método directo y activo. Se caracteriza porque su análisis se realiza sobre los datos obtenidos directamente de las imágenes. Es activo porque hay que emplear algún sistema generador de luz, por lo que introduce un tipo de energía al entorno donde se realiza el estudio.

Los sistemas de luz estructurada se basan en determinar la deformación que sufre un patrón de luz al ser intersecado por cualquier objeto. Para ello se necesita de un tipo de luz concentrada en una región determinada. No serviría como sistema de iluminación cualquiera de los sistemas normales que se emplean actualmente como bombillas, fluorescentes, etc., ya que están compuestos por ondas de diferentes frecuencias, provocando que el haz se difumine por todo el entorno.

Una de las mejores soluciones es emplear un haz láser, debido a sus características de coherencia y direccionalidad. Estas características permiten que todas las ondas del haz tengan la misma frecuencia y que se dirijan a cualquier región.

El emplear puntos láser implica tener que recorrer el objeto por toda su superficie tomando una gran cantidad de puntos y pudiendo perder algunas zonas de éste.

El empleo de un plano es una mejor solución que la anterior. El plano iluminará un conjunto de puntos con las mismas características y que cumplen con la condición de un plano en el espacio¹².

¹² www.elai.upm.es/spain/Investiga/GCII/areas/luz/luz%20estructurada.htm

5.1.1 Láser de línea.

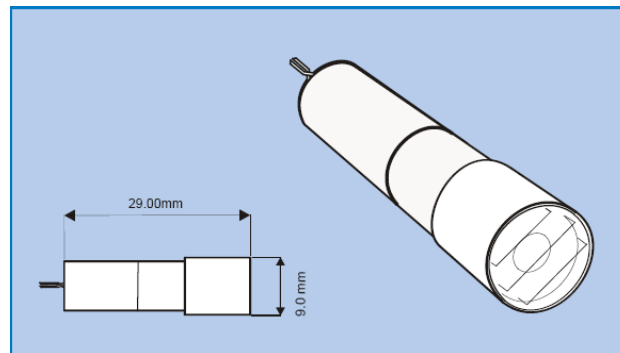


Fig. 14, Dimensiones del láser.

Se utilizó un láser de línea de referencia 300-0238-00 de PHOTONIC Products, con longitud de onda de 635nm, un ángulo de proyección de 88° y potencia de salida de 4mw.

Opera a un voltaje de 3v a 6v DC, una corriente típica de 40mA y una corriente máxima de 50mA. El rango de temperatura va de -10°C a +40°C.

Estas y otras especificaciones técnicas se presentan en la parte derecha de la siguiente tabla:

Specifications (typical @ tc = 25°C)		
Part no.	300-0235-00	300-0238-00
Wavelength	635nm typ., 640nm max	635nm typ., 640nm max
Output Power	0.9mW ±5% @ 25°C	4mW ±5% @ 25°C
Power Stability	< 5%	< 5%
Output Structure	line 88° full angle	line 88° full angle
Focus Distance (other distance can be set to order)	1m	1m
Operating Voltage (DC)	3.0-6.0V	3.0-6.0V
CW Operating Current	30mA typ., 45mA max	40mA typ., 50mA max
Operating Temperature (non-condensing)	-10°C to +40°C	-10°C to +40°C
Storage Temperature	-40°C to +85°C	-40°C to +85°C
Length	29mm	29mm
Diameter	8mm (9mm for lens cap)	8mm (9mm for lens cap)
Housing Material	Brass	Brass
Flying Lead Length	300mm	300mm
Circuit Protection	Static, Surge and Reverse Polarity Protected	

Tabla 3, Especificaciones del láser

5.1.2 Seguridad Visual. El estándar IEC60825 clasifica los productos láser en tres categorías dependiendo de la luz emitida, longitud de onda y seguridad visual.



CLASE II

Precaución, láser de luz visible menor a 1mw. Se considera seguro para los ojos. Si hay una exposición normal a este tipo de rayos, no causará daño permanente a la retina.

CLASE III R

Peligro, láser de luz visible entre 1mw y 5mw. Si se enfoca a los ojos podría causar algún daño.

CLASE III B

Peligro, infrarrojo (IR) y luz láser visible de alta potencia considerada muy peligrosa para la retina.

5.1.3 Distribución de Intensidad.

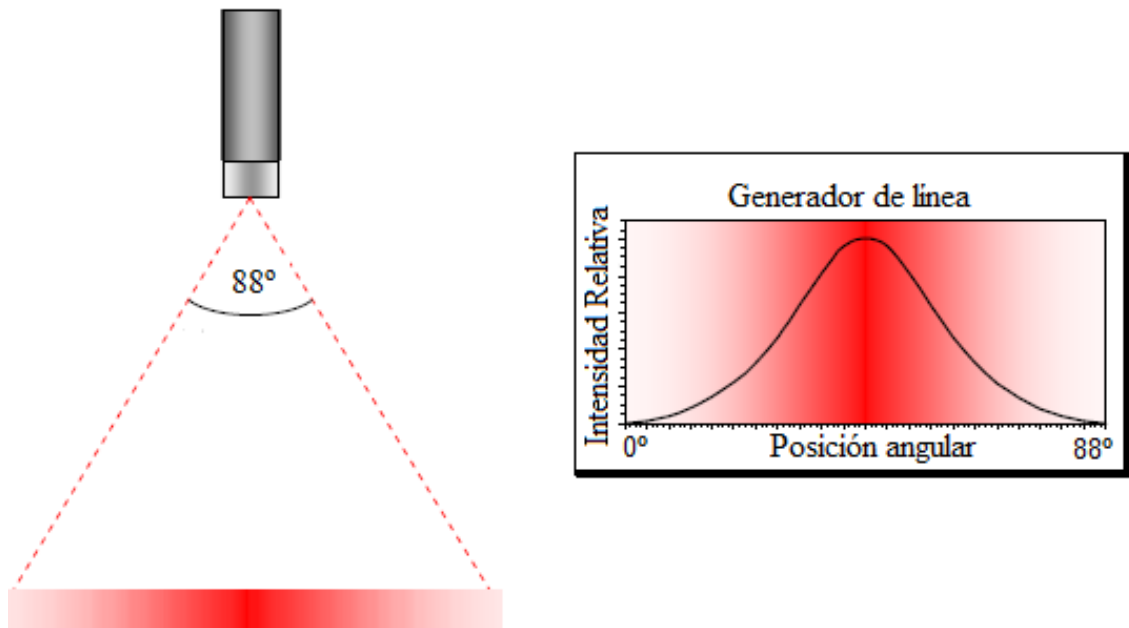


Fig. 15, Ángulo de abertura y distribución de intensidad

El generador produce una línea láser con distribución de intensidad Gaussiana. Esto significa que la línea aparece más brillante en el centro y se opaca en los límites como se muestra en la figura 15. Por este motivo se utiliza la información del centro de la línea, es decir, los píxeles con mayor intensidad luminosa.

5.2 LUZ BLANCA

La luz, al igual que el sonido, es una combinación de "tonos" de diferente frecuencia. Se puede decir que los tonos es al sonido lo que los colores es a la luz. La luz es entonces una combinación de colores (cada color de diferente frecuencia y longitud de onda).

La luz blanca es la luz natural, reflejada por las nubes en horas próximas al mediodía o la que resulta de alguna otra composición similar. Con cielo azul, la luz es un tanto azulada.

La luz blanca es la resultante de las diferentes longitudes de onda que integran el espectro solar. Cada uno de estos rayos tiene su propia longitud de onda, y es la variación de esta longitud de onda la que permite obtener todos los colores posibles.

5.2.1 Disposición de los leds de luz blanca.

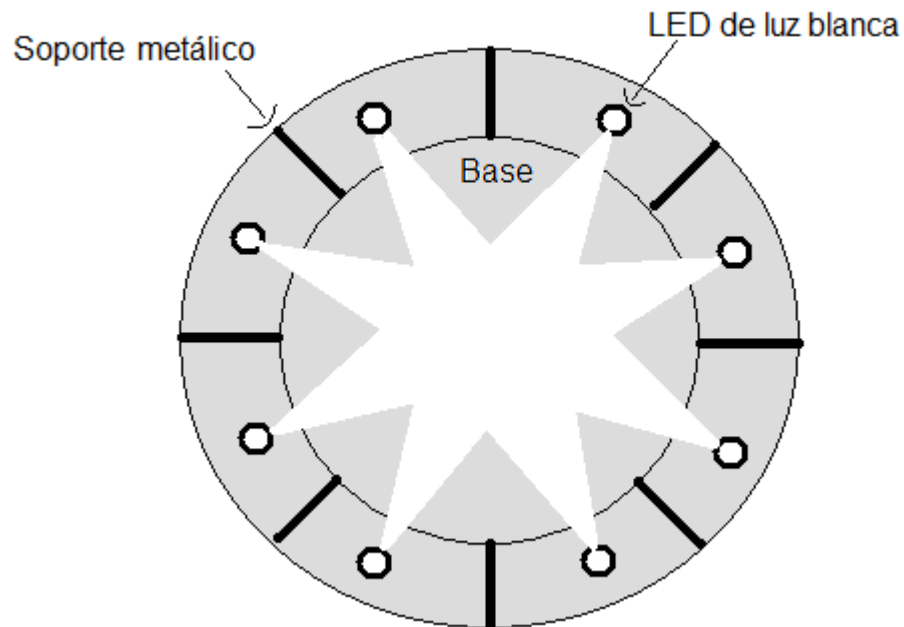


Fig. 16, Ubicación de los leds de luz blanca.

Los leds de luz blanca funcionan a 12v y consumen 18mA cada uno. Los rayos que generan se distribuyen en forma cónica, iluminando una parte del objeto (figura 16). Para una iluminación completa se utilizan 8 leds, separados uno de otro 45° y a 25cm del centro de la base.

5.3 CÁMARA

El dispositivo CMOS es un chip de silicio sensible a la luz que consiste en una matriz de diodos independientes llamados píxeles (*picture element*), los cuales captan los fotones (elementos que componen la luz). A mayor intensidad de luz, más carga eléctrica existirá.

Estos fotones desprenden electrones dentro del sensor de imagen, los cuales se transformarán en una serie de valores (datos digitales) creando un píxel, el cual formará un punto en la imagen.

“Los sensores de imagen CMOS se han diseñado con la capacidad de integrar un número de funciones de procesamiento y control directamente en el circuito integrado del sensor, lo que se extiende más allá de la tarea fundamental de recopilación de fotones. Estos nuevos aspectos incluyen generalmente lógica temporal, control de exposición, conversión de analógico a digital, obturación, balance de blancos, ajuste del aumento y algoritmos de procesamiento inicial de la imagen”¹³.

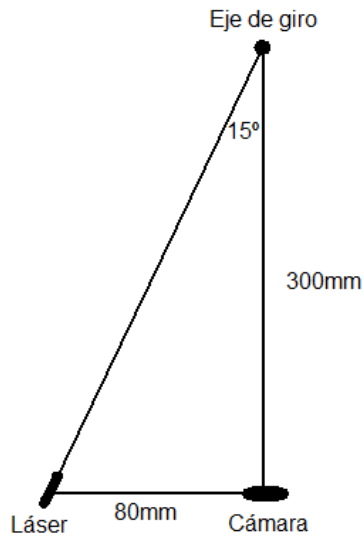
Se utilizó una cámara web de GENIUS con las siguientes características:

VideoCAM Messenger	
Plataforma	Portátiles u ordenadores
Interfaz	USB 1.1
Tipo de sensor de imagen	640x480, 300K píxels
Imagen fija	Interpolación hasta 1.3M pixel
Tipo de lente	Enfoque manual
Máx profundidad de color	32 bits
GIF	30 fps
VGA	12-15 imágenes por segundo
Compresiones AVI	Si
En seguimiento de normativa TWAIN	Si
Dimensiones (LarxAxAI mm)	31,5 x 81 x 59,5
Tipo de sensor de imagen	VGA (640 x 480) , Sensor de imagen CMOS
Tasa máxima de captura AVI	30 fps

Tabla 4, Especificaciones de la cámara

¹³ www.digitalfotored.com/imagendigital/sensorimagen.htm

5.3.1 Disposición de la cámara y el láser.



El ángulo entre el eje óptico de la cámara y el plano láser es de 15° y la separación entre los dos dispositivos es de 80 mm. La distancia entre el eje de giro del eslabón y la cámara es 300mm. La figura 17, muestra la ubicación de estos elementos.

Fig. 17, Ubicación de la cámara y el láser

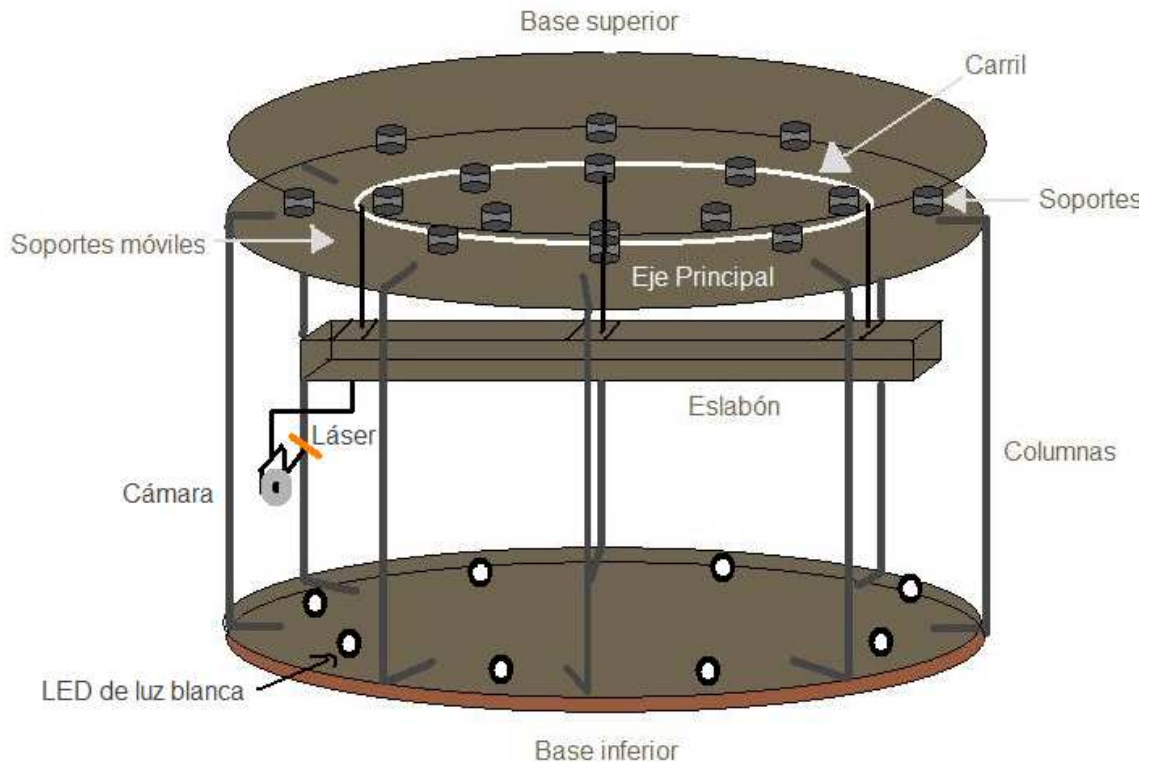


Fig. 18, Estructura mecánica.

La figura 18, representa la estructura mecánica del digitalizador tridimensional. Sobre la base inferior se dispone el objeto a ser digitalizado. El eslabón gira alrededor del eje principal realizando el escaneo y está acoplado a un sistema de engranajes movidos por un motor paso a paso (el motor y los engranajes no se muestran en la figura). La estructura tiene un recubrimiento en poliestireno que aísla lumínicamente el interior del sistema.

El eslabón posiciona la cámara y el láser alrededor del objeto y también posee un sistema de desplazamiento radial, para lo cual utiliza otro motor paso a paso como muestra la figura 19. Este desplazamiento solo se utiliza en el proceso de calibración.

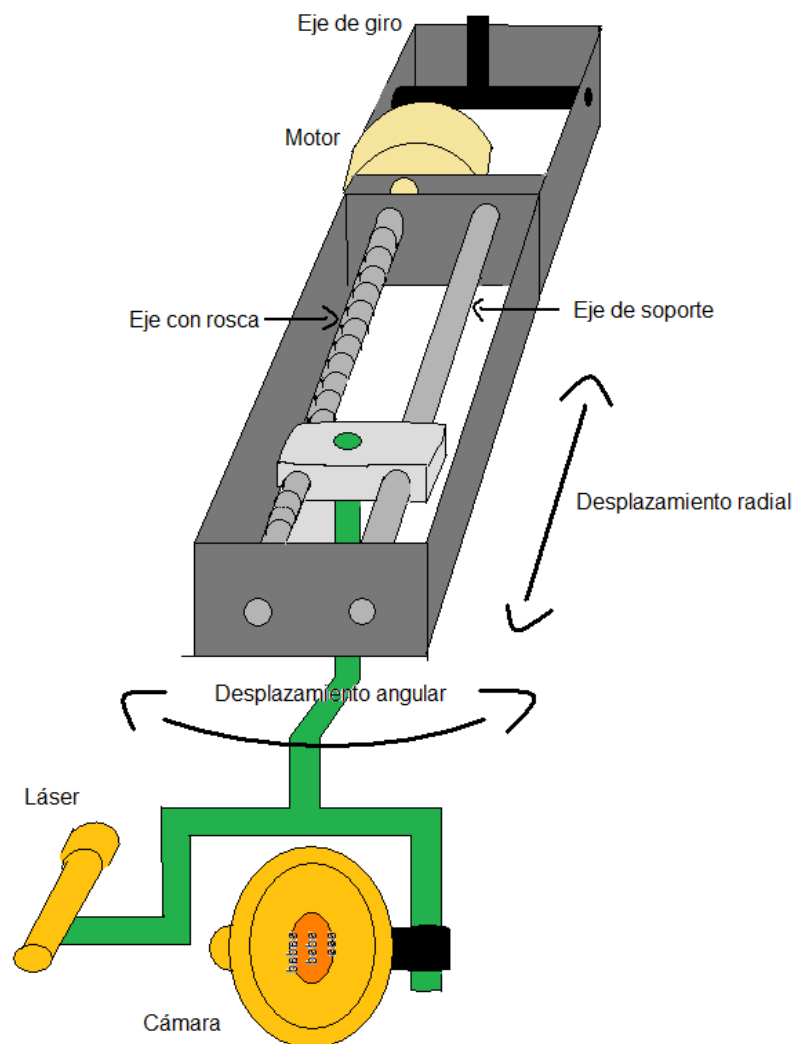


Fig. 19, Eslabón, cámara y láser.

6. SISTEMA ELECTRÓNICO

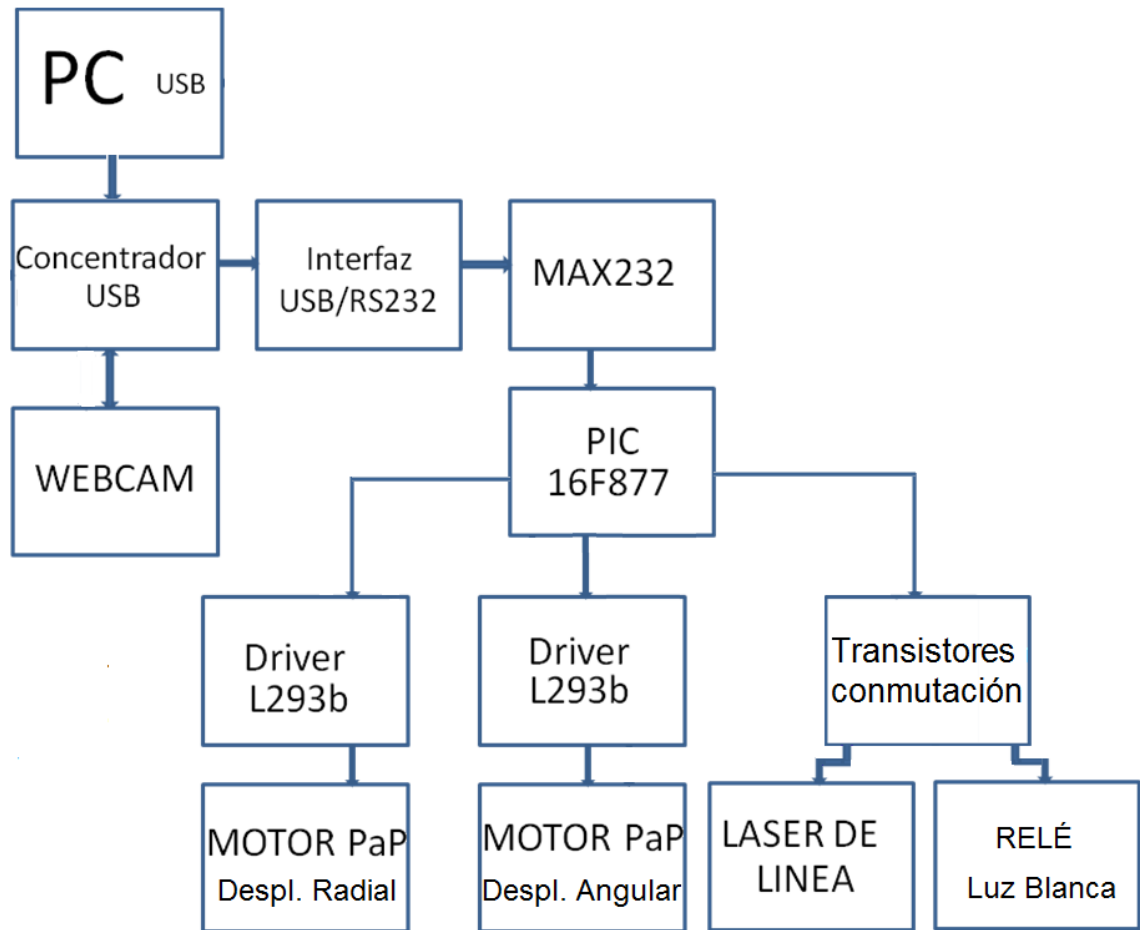


Fig. 20, Diagrama de bloques del sistema electrónico.

El sistema electrónico del digitalizador es el encargado de recibir y decodificar las órdenes del computador y de que todos sus componentes actúen de forma determinada en el proceso de obtención de imágenes. Este sistema está compuesto por diferentes módulos:

6.1 MÓDULO DE CONTROL

- PIC 16F877
- MAX232

El módulo de control recibe las órdenes del computador por medio del puerto serial. Estos datos son transmitidos a una velocidad de 2.4Kbps (8N1) y son recibidos y decodificados por el MAX232 de modo que el PIC 16F877 los procesa y envía información según sea necesario a dos transistores encargados de la conmutación del láser y de la fuente de luz blanca, así como a dos motores paso a paso encargados del movimiento circular del eslabón y radial de la cámara y el láser.

6.2 MÓDULO DE DESPLAZAMIENTO

- Motores Paso a Paso
- Engranajes
- Eslabón
- L293b

El módulo de desplazamiento angular es el que se encarga de que la webcam y el láser se desplacen alrededor del objeto para que realicen los procesos de captura de imágenes y de iluminación. El motor paso a paso se encuentra acoplado mecánicamente a un sistema de engranajes los cuales le ayudan a aumentar su fuerza en un orden de nueve, logrando el movimiento angular del eslabón sobre el que se encuentra la cámara y el láser.

El eslabón giratorio posee un mecanismo de desplazamiento radial, compuesto por otro motor paso a paso y un tornillo sin fin. Éste desplazamiento es necesario en el proceso de calibración.

El control del módulo de desplazamiento angular y radial se hace por medio del PIC 16F877, el cual envía las órdenes del computador al L293b. Este driver permite que las órdenes transmitidas desde el microcontrolador tengan la suficiente potencia para manejar los motores paso a paso.

6.3 MÓDULO DE SENSADO

- Láser de Línea
- Luz blanca
- Cámara Web

Este módulo es el encargado de tomar las imágenes del objeto con y sin línea láser. La webcam recibe las ordenes directamente del computador, mientras que el láser de línea recibe las órdenes por medio del módulo de control.

Lo que hace el computador es enviar la orden a la webcam para que tome una fotografía del objeto. Luego de un tiempo de refresco, se ordena al láser que proyecte el patrón de luz y así, la webcam tome una segunda fotografía. Al obtener estas dos imágenes, el computador se encarga del procesamiento que dará como resultado las coordenadas espaciales de este segmento del cuerpo.

Luego de la captura de las dos imágenes, se produce un desplazamiento angular del eslabón que sostiene a la cámara y al láser para repetir el proceso hasta realizar un escaneo de 360°. Este primer escaneo permite adquirir información geométrica de la superficie del objeto que fue iluminada por la línea láser.

En este punto se activa la fuente de luz blanca que ilumina al objeto para la posterior adquisición del color. Para tal fin se realiza un escaneo en sentido contrario hasta que el eslabón llegue a su punto de partida.

6.4 CIRCUITO ELECTRÓNICO

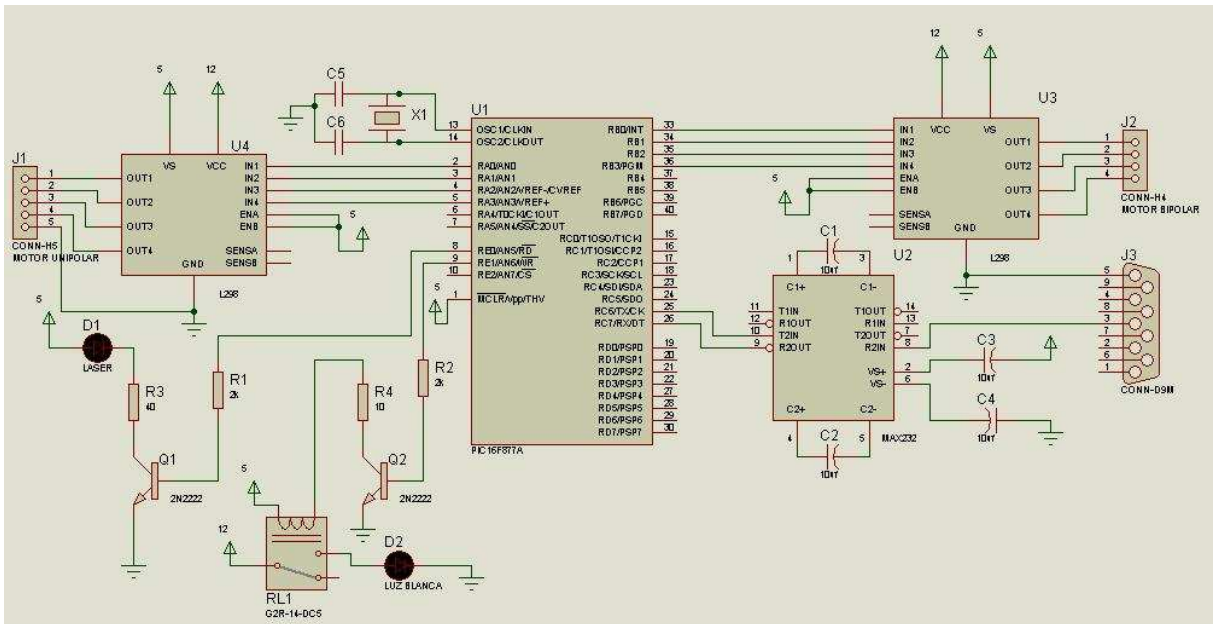


Fig. 21, Circuito electrónico.

El circuito está conformado por la interfaz USB/serial que permite la comunicación serial utilizando el puerto usb de una laptop, el max232 que acondiciona los niveles de voltaje, el microcontrolador PIC16F877 que decodifica las órdenes que envía el computador, dos drivers l293b para cada uno de los motores paso a paso, el láser y la luz blanca.

7. ESCANEEO DEL OBJETO

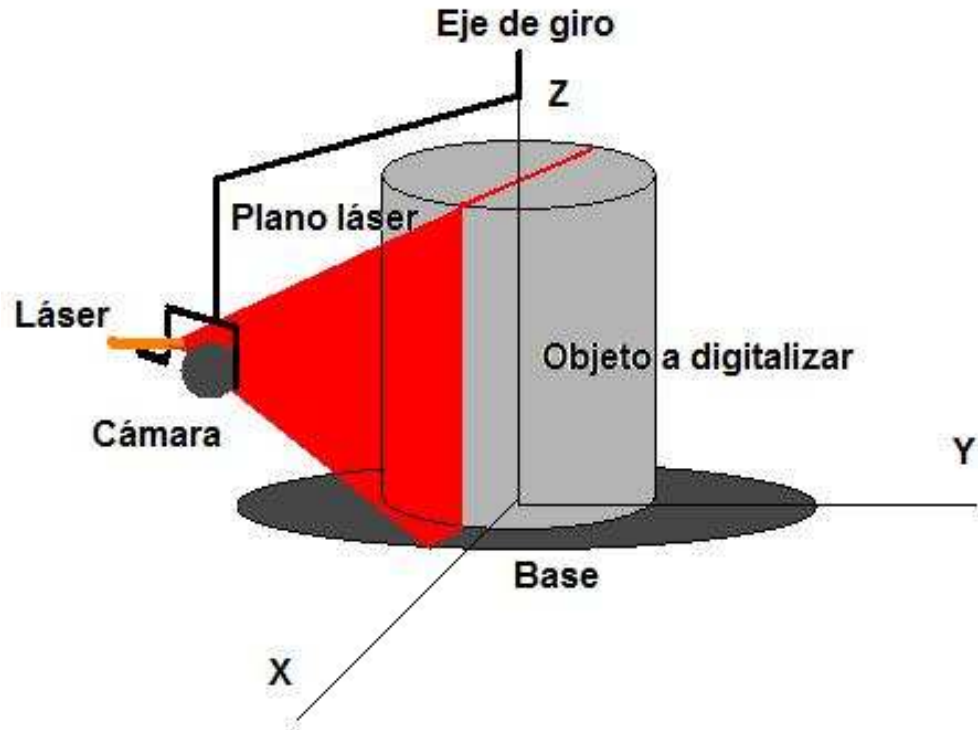


Fig. 22, Escaneo del objeto

El conjunto cámara láser se encuentra en el extremo de un eslabón a 300mm del centro de la base y gira alrededor del objeto, posicionándose cada θ grados (que corresponde al incremento en el desplazamiento angular) para tomar dos imágenes, una con el patrón de luz proyectado en la superficie del objeto y otra sin éste. El barrido del plano láser alrededor del objeto se indica en la figura 22.

El plano láser debe pasar por el centro del sistema o eje z, que corresponde al eje de giro del eslabón o el centro geométrico de la base, cuando no hay ningún objeto sobre ésta.

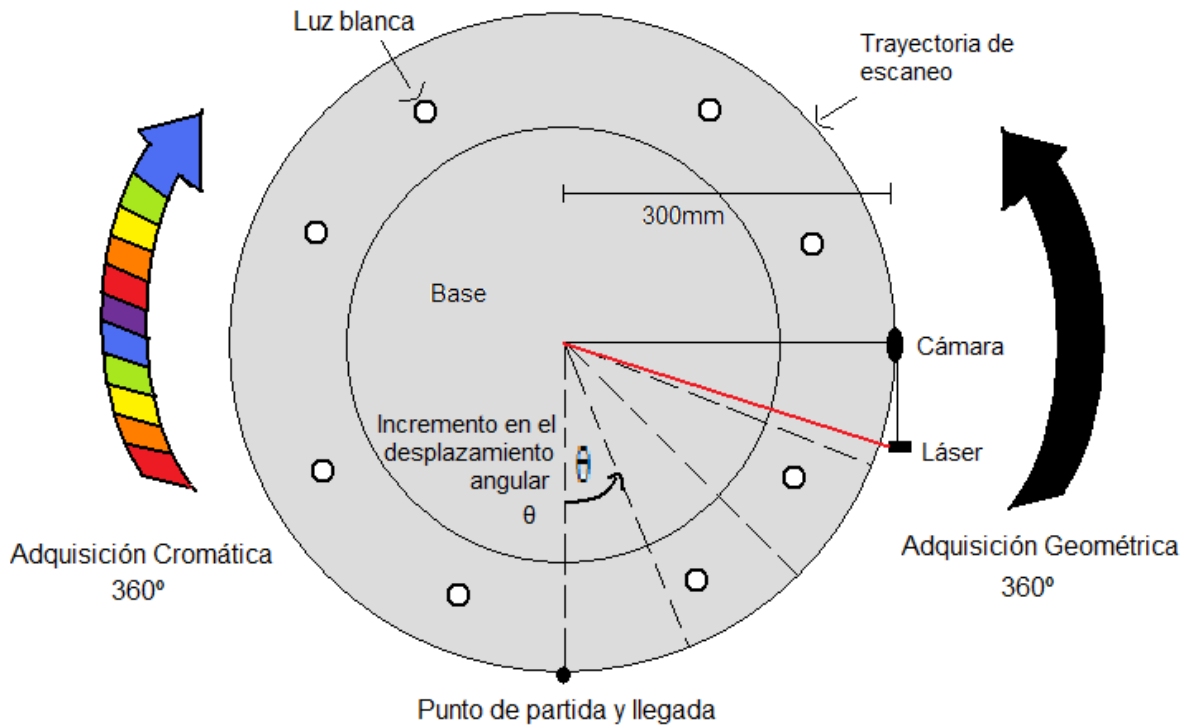


Fig. 23, Vista superior del proceso de escaneo

Por medio de la interfaz gráfica se selecciona el incremento en el desplazamiento angular θ para cada adquisición de imágenes (incremento mostrado en la figura 23). Se puede seleccionar θ entre tres opciones: 1° (360 perfiles) 1.2° (300 perfiles) y 1.8° (200 perfiles), determinando éste valor la cantidad de perfiles obtenidos en un escaneo y la densidad de la nube de puntos.

Luego de realizado el barrido completo del plano láser sobre el objeto, el eslabón que posiciona la cámara y el láser debe regresar a su punto de inicio, para lo cual realiza un giro en sentido contrario. Durante este desplazamiento, se adquieren imágenes del objeto iluminado por la fuente de luz blanca para adquirir la información cromática.

El software realizado se encarga de la adquisición de imágenes, de la conmutación del láser, de la conmutación de la fuente de luz blanca, del posicionamiento de la cámara y el láser y del retorno del eslabón giratorio, además del posterior procesamiento de imágenes y la reconstrucción tridimensional del objeto.

8. PREPROCESAMIENTO DE IMÁGENES

8.1 SUSTRACCIÓN DE IMÁGENES

Para cada perfil se toman dos imágenes, una con patrón de luz y otra sin éste. Luego se sustrae elemento por elemento, siendo ambas imágenes de igual tamaño. Esta operación aritmética genera una imagen del mismo tamaño que las iniciales y de la forma:

$$C(i,j) = A(i,j) - B(i,j) \quad (10)$$

Lo cual indica que cada elemento de C es la diferencia de los correspondientes elementos de A y B.

El comando utilizado es *imabsdiff* de Matlab que computa la diferencia absoluta entre dos imágenes.



Fig. 24, Línea láser deformada por el objeto.



Fig. 25, Imagen del objeto sin proyección de la línea láser.



Fig. 26, Resultado de la sustracción de imágenes

La diferencia entre las dos imágenes reduce el ruido de fondo producido por la iluminación del ambiente debido a que zonas iluminadas por alguna fuente de luz externa aparecerán en ambas imágenes y se restarán, quedando en la imagen resultante la proyección de la línea sobre el objeto.

8.2 CONVERSIÓN DE RGB A ESCALA DE GRISES



Fig. 27, Imagen RGB



Fig. 28, Imagen en escala de grises

Inicialmente se trabajó con la trama de rojo de las imágenes RGB, pero se encontró que las intensidades más altas de rojo no correspondían a la línea láser sino a reflexiones secundarias sobre el objeto y que la línea láser era blanca y por tanto compuesta por fracciones similares de rojo, verde y azul. Por tal motivo se utilizó imágenes en escala de grises como la que se indica en la figura 28.

Matlab permite hacer la conversión de imágenes a escala de grises con el comando *rgb2gray*.

8.3 TRANSFORMACIÓN DE INTENSIDAD

Las imágenes en escala de grises se caracterizan por tener un histograma con dos zonas bien definidas. La primera zona la conforman los píxeles con baja intensidad y la segunda, los píxeles de alta intensidad. Para hacer un mayor contraste entre intensidades bajas e intensidades altas (zona de interés) se realiza una transformación de intensidad.

La siguiente ecuación indica una transformación:

$$g(x,y) = T[f(x,y)] \quad (11)$$

Donde el valor de g en (x,y) depende solamente de la intensidad de f en ese punto y T es un operador que representa la función de transformación de intensidad o de niveles de grises.

Debido a que esta transformación depende solo de valores de intensidad y no explícitamente de (x,y) , las funciones de transformación de intensidad frecuentemente son escritas en forma simplificada, como se indica a continuación:

$$s = T(r) \quad (12)$$

Donde r denota la intensidad de f y s la intensidad de g , ambos en cualquier punto (x,y) de la imagen.

La función *imadjust* de Matlab es una herramienta de procesamiento de imágenes utilizada para transformaciones de intensidad de imágenes en escala de grises.

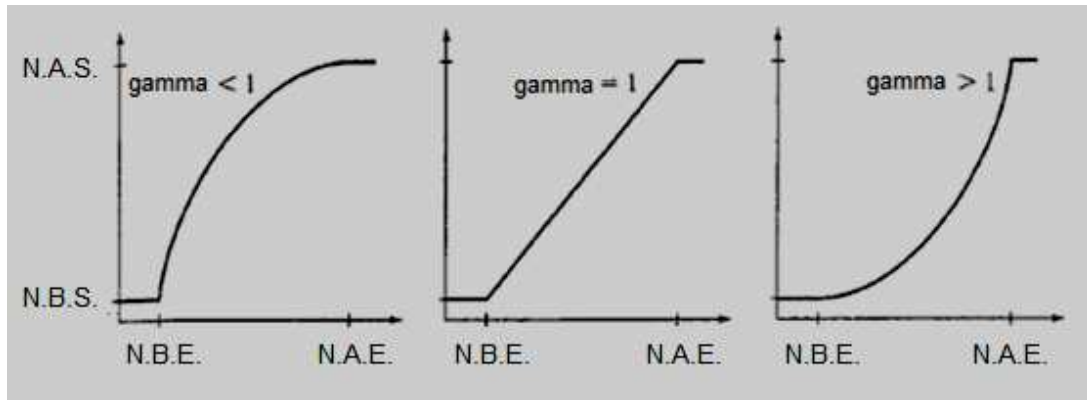


Fig. 29, Transformación de intensidad utilizando imadjust (nivel alto N.A y bajo N.B de entrada y salida).

Como se muestra en la figura 29, “ésta función mapea los valores de intensidad de la imagen f a nuevos valores g , tal que valores entre N.B.E. y N.A.E. mapean valores entre N.B.S. y N.A.S. Los valores de entrada que se encuentran por debajo de N.B.E. y por encima de N.A.E. se les asignan a la salida el valor de intensidad bajo y alto respectivamente.

El parámetro γ especifica la forma de la curva que mapea los valores de intensidad de f para crear g . Si γ es menor que 1, se realzan los valores bajos de intensidad de entrada hacia valores más altos. Si γ es mayor que 1, se reducen los valores bajos de intensidad de entrada a valores más bajos. Si γ es 1, el mapeo es lineal”¹⁴.



Fig. 30, Imagen inicial.

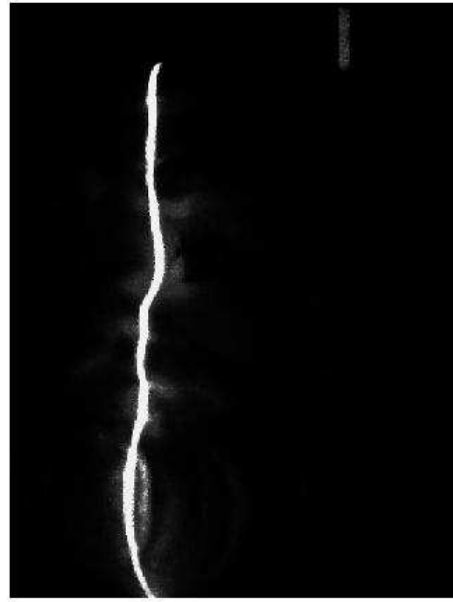


Fig. 31, Resultado de la transformación de intensidad.

¹⁴ GONZALEZ, Rafael. *Digital Image Processing Using Matlab*. Prentice Hall. Capítulos 2,3,9,10.

La figura 31 muestra el resultado de aplicar a la imagen de la figura 30 la función *imadjust* con N.B.E = 0, N.A.E = 1, N.B.S = 0, N.A.S = 1 y gamma = 2 para reducir los valores de baja intensidad.

Los histogramas se obtuvieron con el comando *imhist* de Matlab.

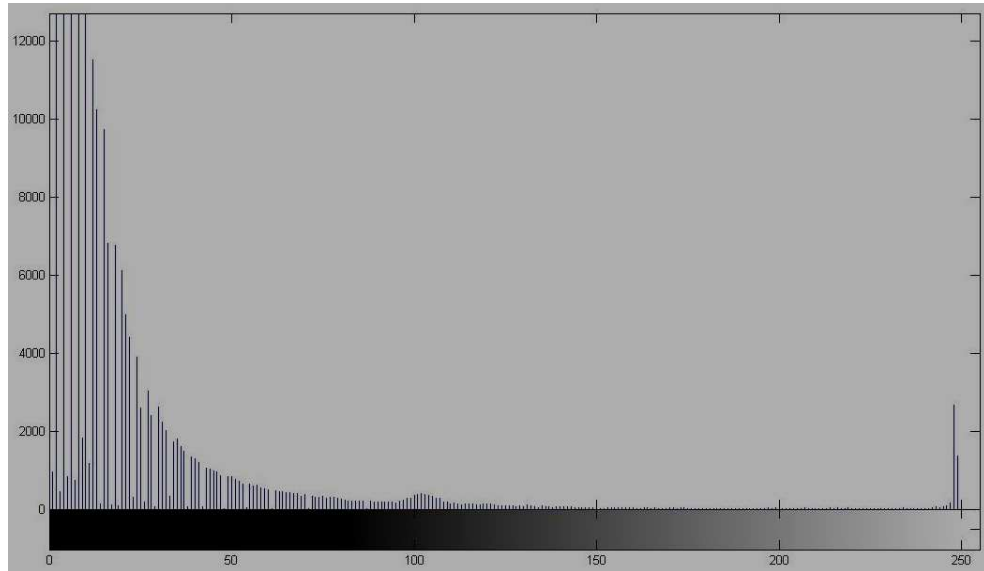


Fig. 32, Histograma de la imagen inicial (mostrada en la figura 30).

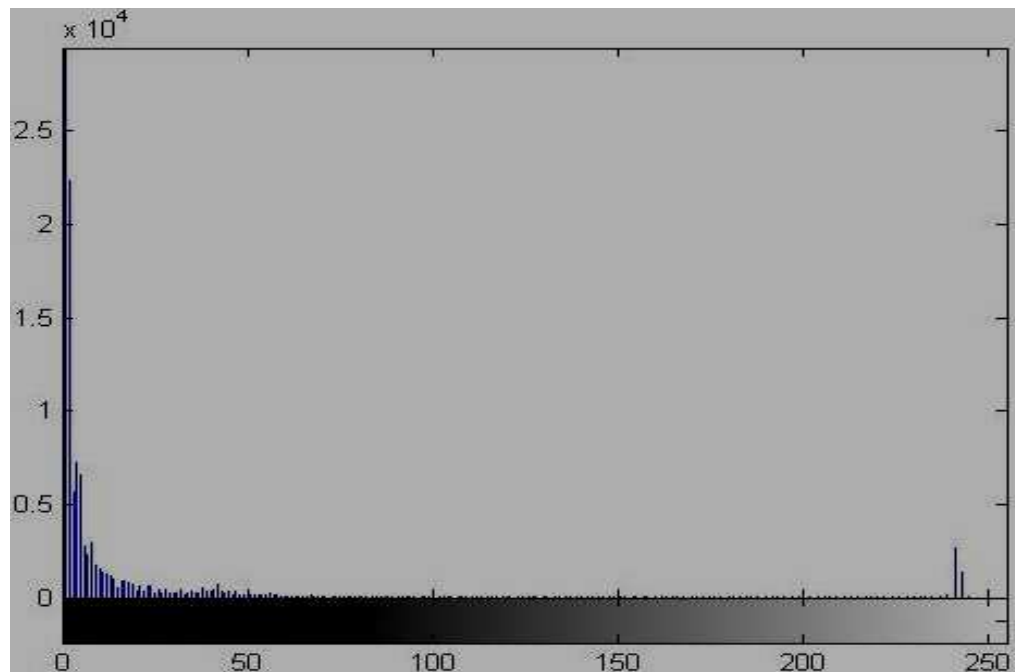


Fig. 33, Histograma de la imagen con transformación de intensidad (mostrada en la figura 31).

8.4 TRANSFORMACIÓN DE UMBRAL

“La función de transformación de umbral comprime los niveles de entrada más bajos que un umbral m en una banda delgada de niveles oscuros en la imagen de salida. Similarmente, comprime los valores por encima de m en una banda delgada de niveles claros en la imagen de salida. El resultado es una imagen de alto contraste”¹⁵.

La siguiente imagen ilustra su funcionamiento.

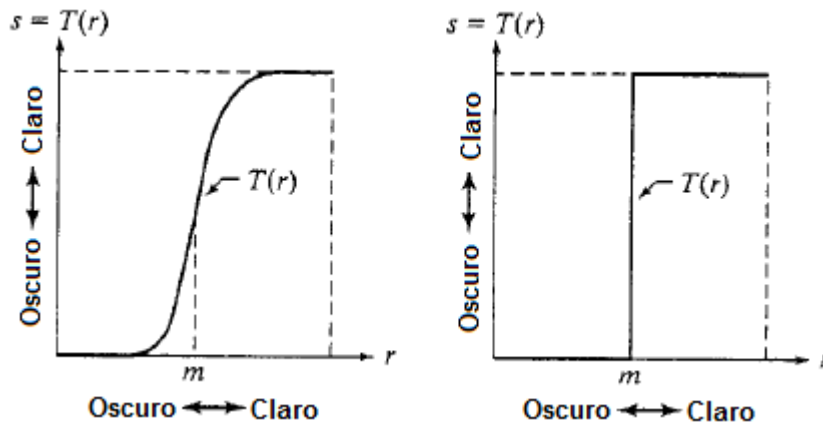


Fig. 34, Transformación de Umbral.

$$s = T(r) = \frac{1}{1 + \left(\frac{m}{r}\right)^E} \quad (13)$$

Donde r representa las intensidades de la imagen de entrada, s las intensidades de la imagen de salida y E controla la inclinación de la función.

Se utilizó una gran inclinación como se muestra en la parte derecha de la figura 34. Para lograrlo, E fue igual a 20. Por debajo del umbral las intensidades tienden a cero y por encima de éste, las intensidades tienden a 1, asemejándose a una imagen binaria.

8.4.1 Determinación del Umbral. El umbral se determina automáticamente y se basa en el análisis del histograma de cada imagen en escala de grises.

El histograma que caracteriza a cada imagen está compuesto por dos zonas, una de niveles oscuros de grises y otra de niveles claros de grises (figura 35).

¹⁵ GONZALEZ, Rafael. *Digital Image Processing Using Matlab*. Prentice Hall. Capítulos 2,3,9,10.

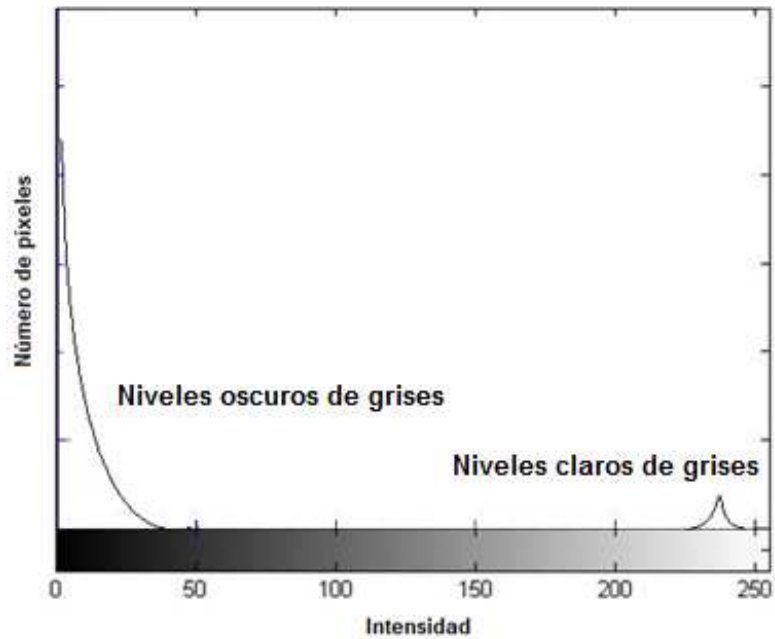


Fig. 35, Histograma característico de las imágenes.

El conjunto de intensidades del lóbulo que están por encima de 150 corresponden a la línea láser y reflexiones de ésta sobre el objeto. El umbral se ubicó un 10% antes del pico máximo de éste lóbulo como se puede observar en la siguiente figura.

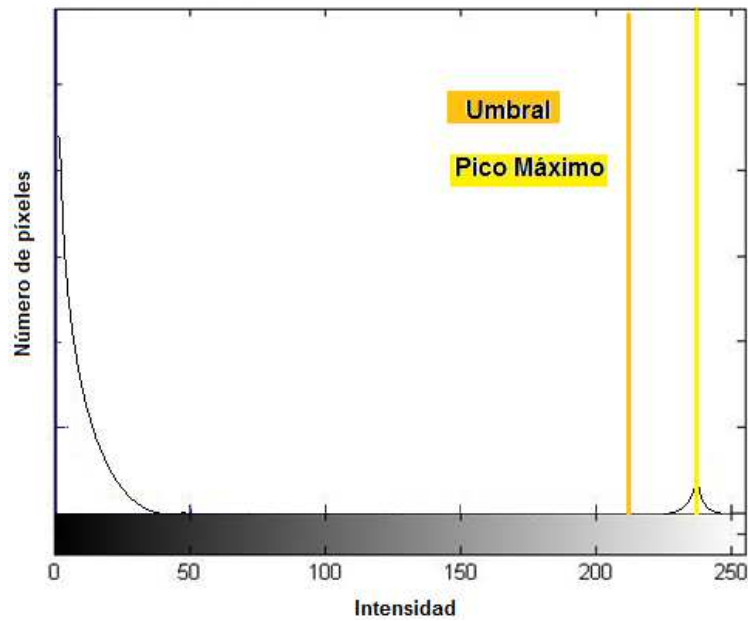


Fig. 36, Umbral ubicado un 10% antes del pico máximo del segundo lóbulo.

$$Umbral = Pico\ Máximo - (0,1 * Pico\ Máximo) = 0,9 * Pico\ Máximo \quad (14)$$

Por ejemplo si el pico máximo del segundo lóbulo es 240, el umbral se ubica a una intensidad de 216.

8.5 IMAGEN BINARIA

Definimos una imagen binaria como una función de dos variables discretas $g[i,j]$ que puede tomar dos valores, '0' (negro) o '1' (blanco), dependiendo de si el nivel de grises de la imagen supera o no el umbral de intensidad T de la siguiente manera:

$$g(i, j) = \begin{cases} 1 & \text{para } f(i, j) \geq T \\ 0 & \text{para } f(i, j) < T \end{cases} \quad (15)$$

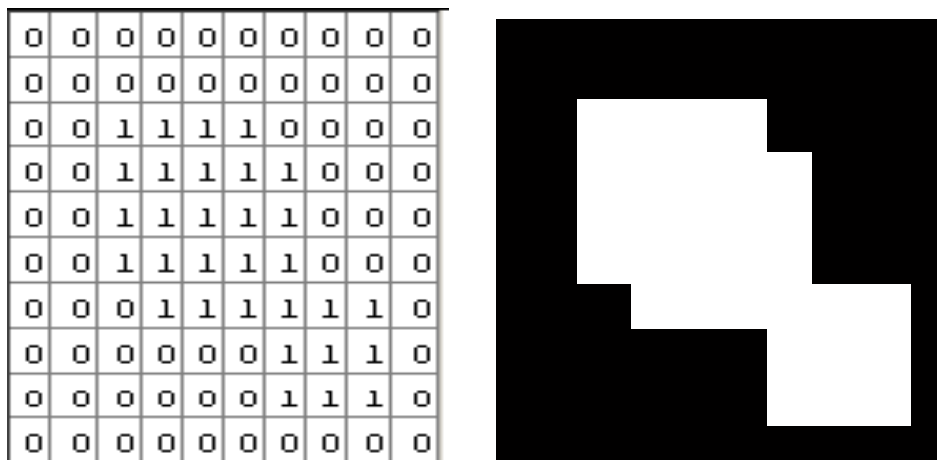


Fig. 37, Ejemplo de imágenes binarias

La función *im2bw* se encarga de convertir los valores obtenidos con la transformación de umbral en ceros y unos, creando una imagen binaria (figura 38).

	153	154	155	156	157
180	0.8837	0.8662	0.0035	5.1101e-16	2.6888e-18
181	0.8837	0.8662	1.4932e-04	5.1805e-18	1.5368e-19
182	0.8837	0.8662	1.0748e-05	9.3226e-24	7.0136e-20
183	0.8990	0.8240	6.2172e-07	9.3754e-29	3.1000e-20
184	0.8990	0.0898	7.7115e-08	2.8988e-22	3.1000e-20
185	0.8990	0.0564	1.8063e-08	2.6888e-18	3.1000e-20
186	0.8990	0.0345	5.9813e-09	3.1000e-20	1.5368e-19
187	0.8990	0.0345	2.4455e-10	2.5643e-24	3.2690e-19
188	0.8990	0.0167	1.1694e-08	1.3234e-20	3.2690e-19
189	0.8990	0.3119	7.7115e-08	8.9411e-15	6.7655e-19
190	0.8990	0.7990	2.9841e-07	1.0928e-13	3.2690e-19

	153	154	155	156	157
180	1	1	0	0	0
181	1	1	0	0	0
182	1	1	0	0	0
183	1	1	0	0	0
184	1	0	0	0	0
185	1	0	0	0	0
186	1	0	0	0	0
187	1	0	0	0	0
188	1	0	0	0	0
189	1	0	0	0	0
190	1	1	0	0	0

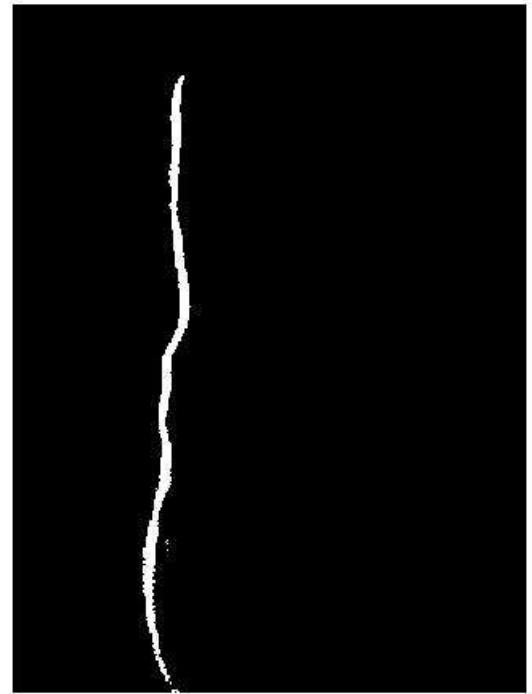


Fig. 38, Conversión a imagen binaria

9. PROCESAMIENTO DE IMÁGENES

Luego del pre-procesamiento de imágenes se pasó a aplicar los operadores y algoritmos morfológicos como se muestra a continuación:

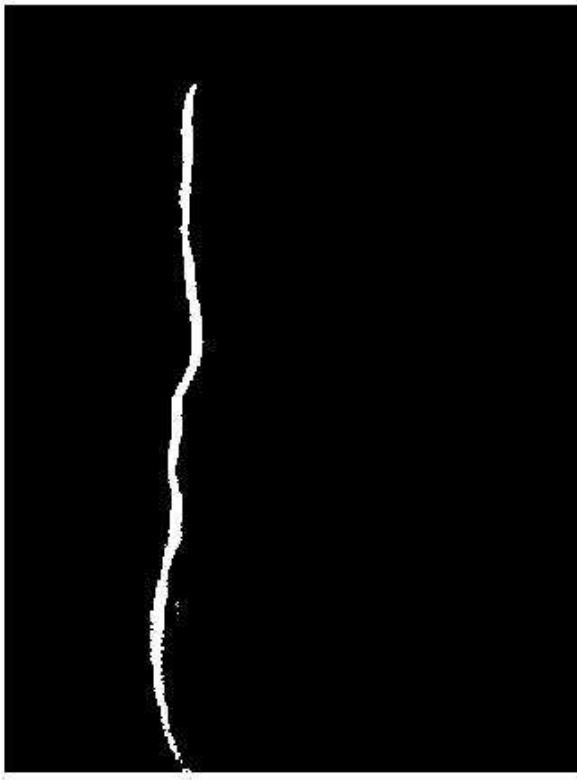


Figura 39. Imagen binaria inicial

Es la que se obtiene con la transformación de umbral y a partir de ésta, se realizan operaciones morfológicas con el fin de mejorar el resultado de la esqueletización y por tanto, de las coordenadas de los puntos que la conforman. La figura muestra una línea con frontera pixelada o no uniforme y unos pequeños conjuntos de píxeles de valor uno que corresponden a reflexiones de la proyección láser sobre el objeto y cuya intensidad superó el umbral.

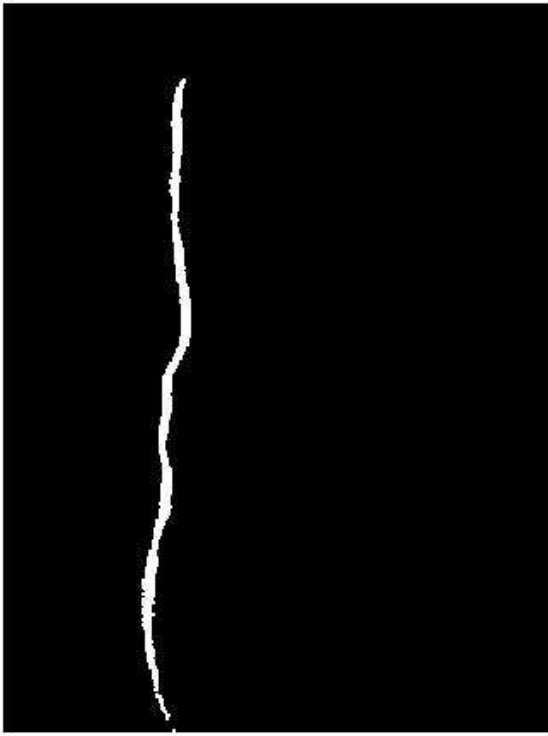


Figura 40. Imagen con apertura

La figura es el resultado de aplicar la operación morfológica de 'apertura' con el comando *imopen*. El elemento estructurante es un cuadrado de 3x3 píxeles. Se alisan contornos y se eliminan grupos de píxeles de valor uno donde no quepa el elemento estructurante.

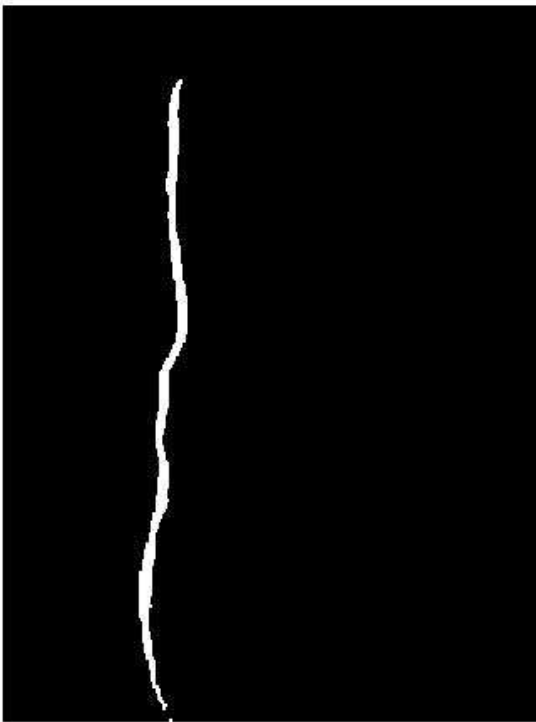


Figura 41. Imagen con cierre

La figura muestra el resultado de aplicar el operador morfológico de 'cierre' con el comando *imclose*. El elemento estructurante es un cuadrado de 8x8 píxeles. Rellena vacios en el contorno.

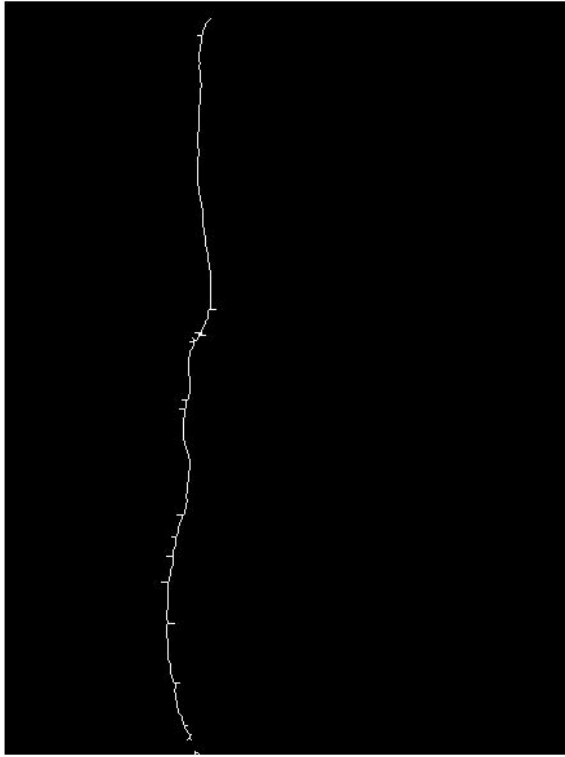


Figura 42. Imagen esqueletizada

La figura es el resultado de aplicar el proceso morfológico de 'esqueletización' con el comando *bwmorph* y *skel*. Se genera una línea de un píxel de ancho y en el centro de la línea de la anterior imagen. Aparecen unas prolongaciones o 'raíces' indeseadas.

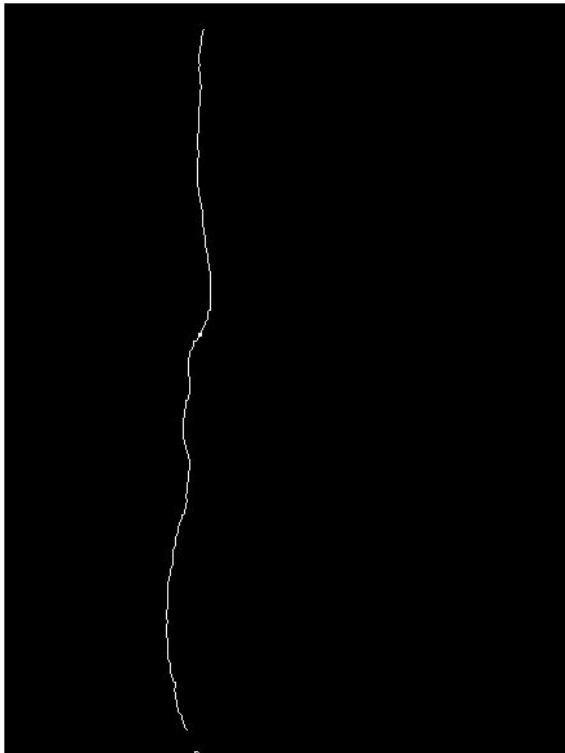


Figura 43. Imagen con poda

La figura es el resultado de aplicar el proceso morfológico de 'poda' con el comando *bwmorph* y *spur*. Se quitan las prolongaciones no deseadas quedando solamente la línea deformada que corresponde a un perfil del objeto.

10. CALIBRACIÓN

Para decodificar la información tridimensional que se encuentra implícita en las imágenes bidimensionales, se recurre a la curva de conversión de unidades, que permite obtener coordenadas en milímetros a partir de coordenadas en píxeles.



Fig. 44, Plano con franjas horizontales

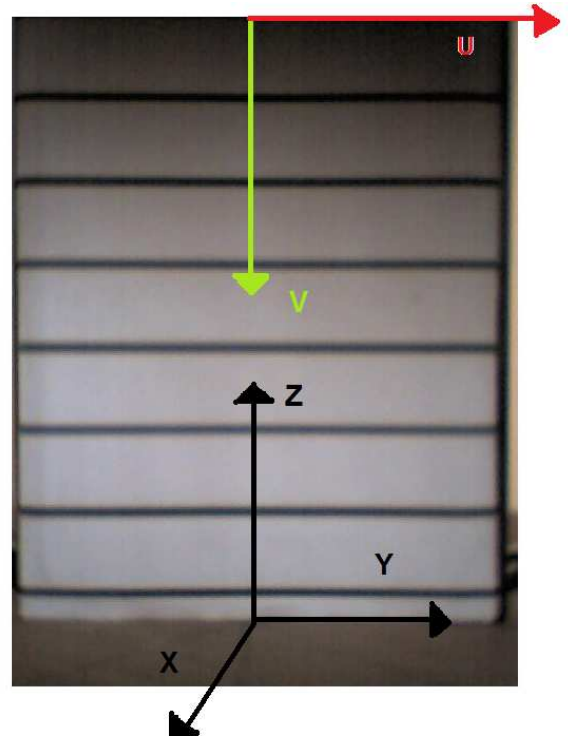


Fig. 45, Plano ubicado sobre el Marco Indicando dirección U y V

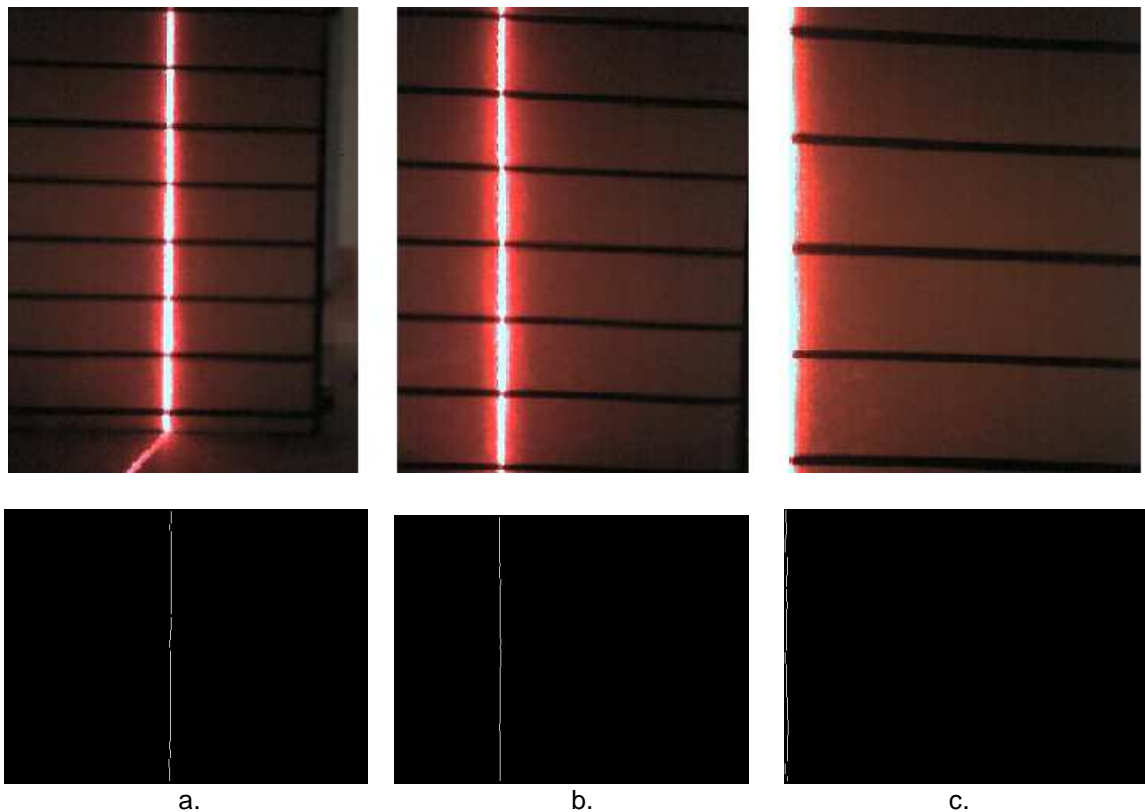
Se utilizó un marco de 220x160mm sobre el que se dispuso un plano con franjas horizontales y paralelas, separadas 27mm. Las franjas se observan en la figura 44 y en la figura 45 aparece el marco ubicado en uno de los diámetros de la base.

Frente al marco se dispuso la cámara y el láser de tal forma que su desplazamiento sea radial y perpendicular al diámetro escogido. En el proceso de escaneo, el conjunto cámara láser gira a una distancia de 300mm del centro de la base. La disposición de todos los elementos y la distancia entre éstos se observa en la figura 46 y 47 respectivamente.

Como las franjas se encuentran sobre una superficie plana, la línea no sufrirá deformaciones, sin embargo se presentan dos fenómenos que permiten la posterior conversión de píxeles a milímetros:

- Desplazamiento en píxeles en dirección U (dirección indicada en la figura 45) que depende del acercamiento entre el marco y la cámara-láser. Se procesan las imágenes para esqueletizar y reducir la línea a una de un píxel de espesor. Luego se obtiene el desplazamiento con respecto a la línea proyectada cuando el conjunto cámara-láser se encuentra a 300mm del centro de la base (línea de referencia).
- Distancia en píxeles en dirección V entre dos líneas horizontales consecutivas (dirección indicada en la figura 45), la cual aumenta a medida que se acerca la cámara-láser al marco. Se procesan las imágenes para encontrar las zonas de la línea con menor intensidad debido a la baja reflexión de la proyección láser sobre las líneas negras.

Estos fenómenos se pueden observar en las siguientes imágenes, donde además se indica el resultado de su procesamiento de esqueletización.



a. b. c.
Fig. 49, Línea láser (parte superior) y esqueletizada (parte inferior);
a. a 300mm del centro b. a 230mm del centro c. a 160mm del centro

Se adquirieron 71 imágenes que corresponden a un desplazamiento total de 140mm. La primera imagen se tomó a una distancia de 300mm, es decir, sin ningún desplazamiento radial y el valor obtenido sirvió como referencia para los demás datos, porque a partir de esta coordenada, la línea empieza su desplazamiento en dirección U.

10.1 CURVAS DE CONVERSION

Hay dos curvas de conversión, según la coordenada de entrada U ó V. Para cada una de ellas, luego de la adquisición de imágenes, se siguen distintos procedimientos como se ve a continuación.

10.1.1 Curva de conversión en U. Para cada una de las 71 imágenes se realizó el siguiente procedimiento:

- Convertir las imágenes RGB a escala de grises.
- Rotar la imagen 90 grados en sentido anti-horario debido a que se dispuso la cámara verticalmente para que pueda enfocar la totalidad del marco.
- Escoger el tamaño de la imagen a procesar debido a que se enfoca una zona que no corresponde al marco.
- Determinar el umbral automáticamente de la imagen para hacerla binaria, es decir, asignar un valor a los píxeles con magnitud que supere el umbral de intensidad y otro a los que se encuentren por debajo de éste ('1' o blanco y '0' o negro respectivamente).
- Esqueletizar o adelgazar la línea deformada para obtener una de un píxel de ancho y que se encuentre en la mitad de la línea sin procesar debido a su distribución de intensidad gaussiana (figura 49).
- Obtener las coordenadas en dirección U de la línea esqueletizada promediando la coordenada de interés de los píxeles que conforman la línea.

El resultado es el siguiente:

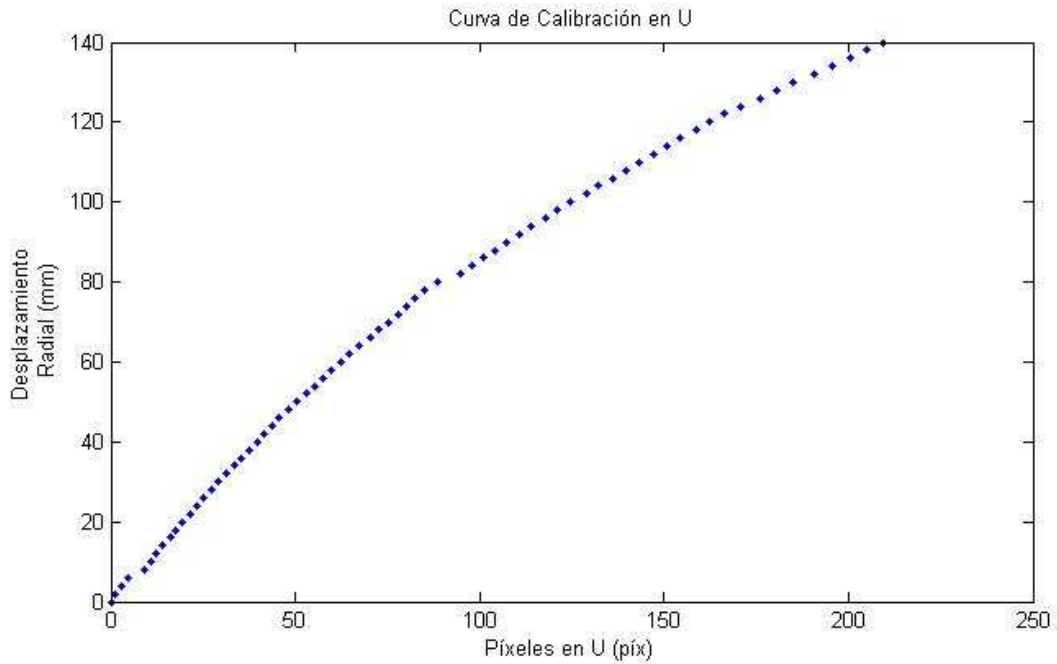


Fig. 50, Gráfica del desplazamiento radial en milímetros y desplazamiento en píxeles con respecto a la línea de referencia y en dirección U

Los datos siguen un comportamiento cuadrático como se puede ver en la siguiente ecuación:

$$Rd = f(U) = -0.0018U^2 + 1.0321U + 1.2921 \quad (16)$$

$$R^2 = 0.999$$

El coeficiente de correlación al cuadrado es muy cercano a la unidad, lo que indica una muy buena aproximación cuadrática.

La función Rd permite determinar la profundidad de un punto conociendo el desplazamiento en dirección U que éste produce sobre la línea láser.

La siguiente imagen (figura 51) muestra la superposición de las 71 imágenes y la mayor cercanía entre líneas en la parte central debido a la relación cuadrática anteriormente descrita.

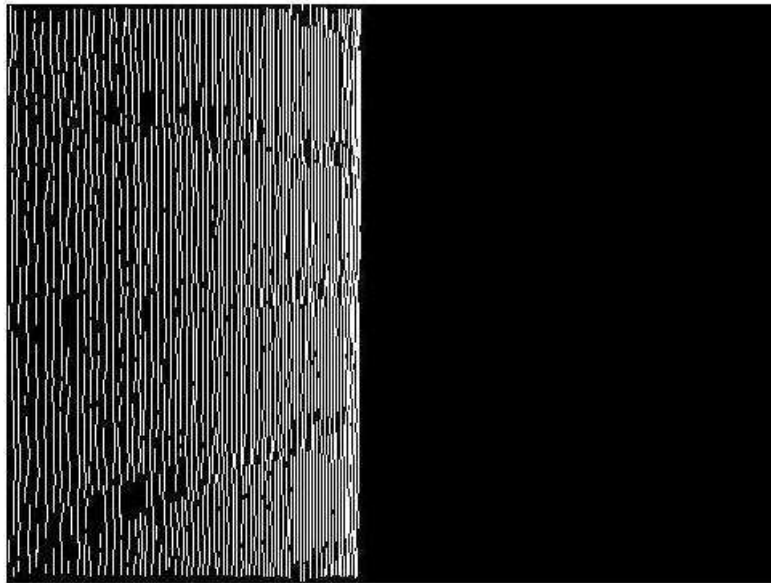


Fig. 51, Superposición de las imágenes con líneas esqueletizadas

* Aplicación de la curva de conversión en U.

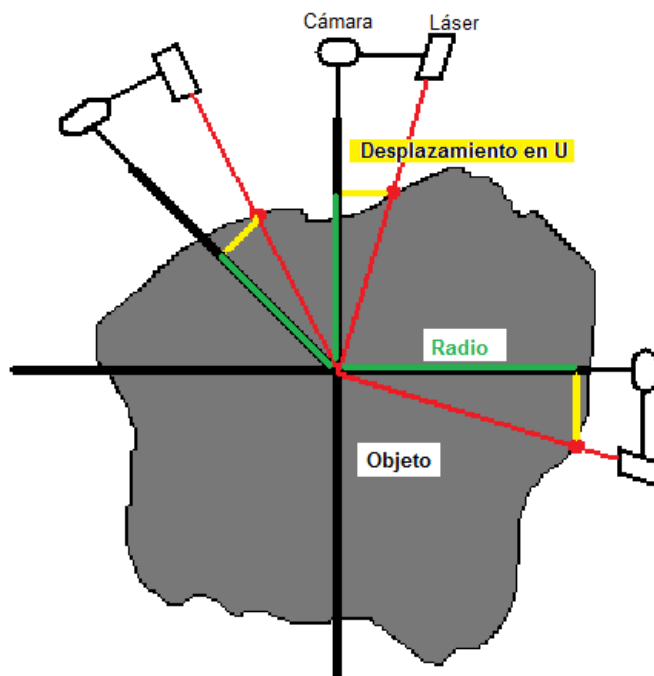


Fig. 52, Vista superior del corte transversal de un objeto, cámara y láser

Para obtener el radio de los puntos que conforma la superficie de un objeto en un plano dado (figura 52), se necesita conocer la separación en píxeles de la línea láser y la línea de referencia. Este valor se utiliza como entrada a la ecuación 16 y se obtiene el radio (en verde) que corresponde a la proyección sobre el eje de la cámara del vector con origen en el centro del sistema y fin en el punto de intersección del plano láser, el plano horizontal y el objeto, como se indica a continuación:

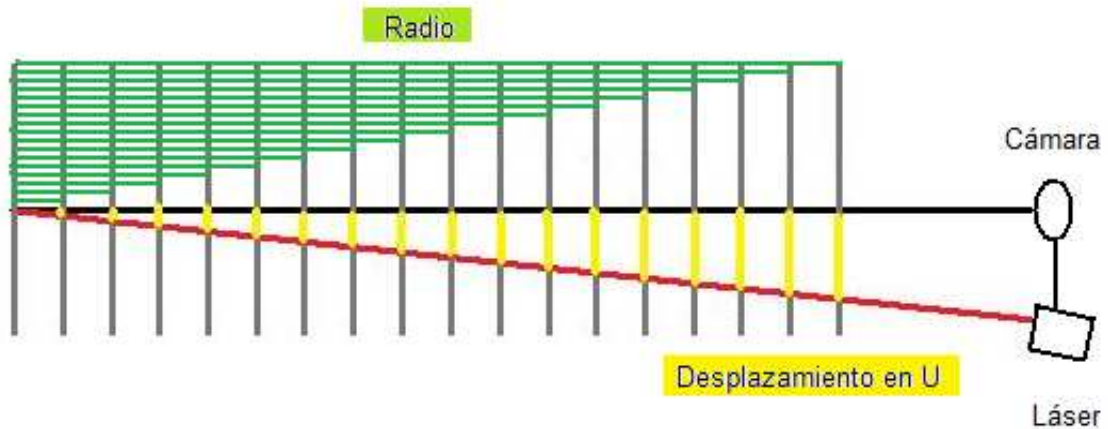


Fig. 53, Descripción gráfica del radio y el desplazamiento en píxeles en dirección U

La ecuación 16 representa la relación entre la magnitud del radio (en verde) y la magnitud en píxeles y en dirección U (en amarillo) de la línea láser y la línea de referencia.

La distancia que se necesita no es el radio sino la que hay desde el centro del sistema y cada punto de iluminación de la línea láser sobre el objeto (la hipotenusa de cada triángulo), la cual es:

$$\rho = \frac{Rd}{\cos 15^\circ} \quad (17)$$

10.1.2 Curva de conversión en V. El procedimiento que se siguió es el siguiente:

- Convertir las imágenes RGB a escala de grises
- Rotar la imagen 90 grados en sentido anti-horario debido a que se dispuso la cámara verticalmente para que pueda enfocar la totalidad del marco.
- Escoger el tamaño de la imagen a procesar (tamaño de la figura 54 y 55).

- Determinar el umbral automáticamente de la imagen para hacerla binaria, es decir, asignar un valor a los píxeles con magnitud que supere el umbral de intensidad y otro a los que se encuentren por debajo de éste ('1' o blanco y '0' o negro respectivamente).
- Ubicar las partes de la línea láser donde su espesor sea menor a 8 píxeles que corresponde a la intersección de ésta con las líneas negras y horizontales del marco, aprovechando la menor reflexión en estas zonas (figura 56).
- Encontrar el centro geométrico de cada zona (figura 57).
- Obtener la diferencia de coordenadas en dirección V entre puntos consecutivos y promediarlos.

Se encontró dos situaciones que prevalecen en la mayoría de las imágenes:

- Dos puntos cercanos sobre la línea negra (figura 57)
- Puntos en los extremos superior e inferior de la imagen (figura 57).

Para evitar estos datos erróneos se realizó un procesamiento adicional en donde se eliminó puntos en los extremos y se promedió datos muy cercanos entre sí (figura 58).

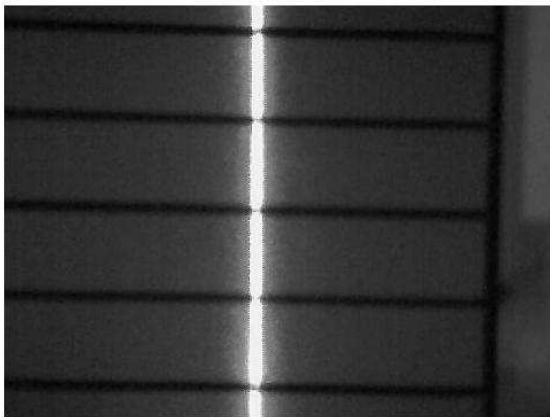


Fig. 54, Imagen en escala de grises

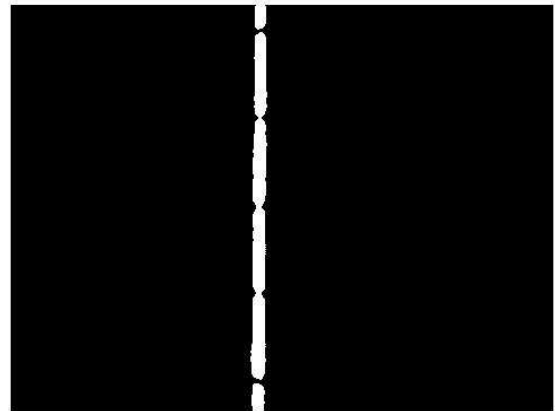


Fig. 55, Imagen binaria con reducción de espesor en la intersección con las líneas negras.

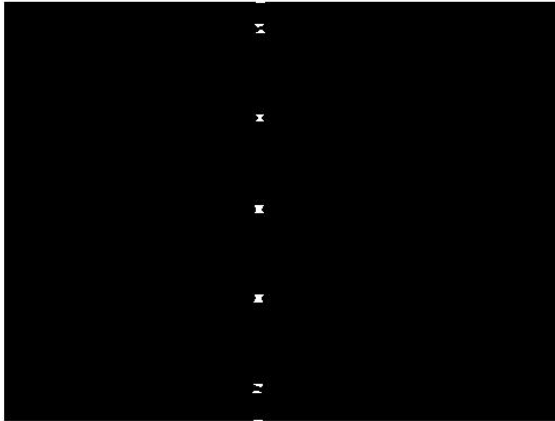


Fig. 56, Zonas de intersección.

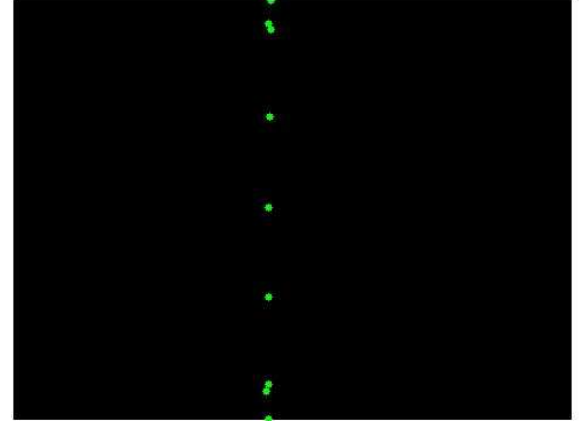


Fig. 57, Centro geométrico de cada zona y errores como puntos muy cercanos y en los extremos.

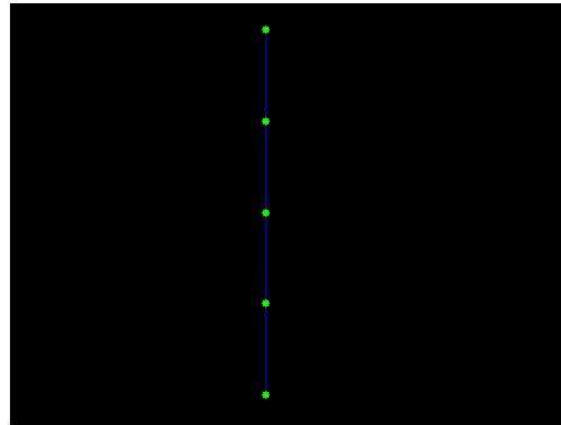


Fig. 58, Corrección de intersección

El factor de conversión es la razón de la separación en milímetros de dos puntos consecutivos (27mm) y la misma separación dada en píxeles (en dirección V). Este último valor depende de la distancia del conjunto cámara láser y el marco, resultando una ecuación de dos variables, U y V.

$$F = \frac{Z}{V} = f(Rd) = f(U); \quad [mm/píxel] \quad (18)$$

$$Z = f(U) * V = f(U, V); \quad [mm] \quad (19)$$

El resultado se puede ver en la figura 59.

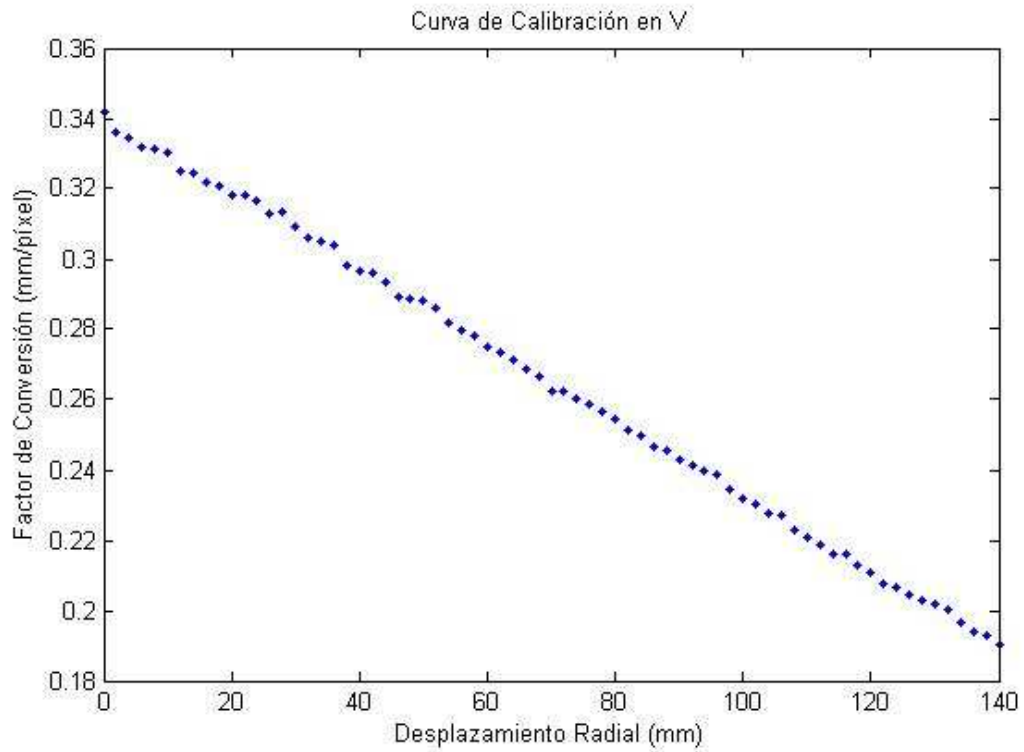


Fig. 59, Gráfica del factor de conversión y el desplazamiento radial.

$$F = \frac{Z}{V} = 2E - 7Rd^2 - 0.0011Rd + 0.3449 \quad (20)$$

$$Z = f(U, V) = (2E - 7Rd^2 - 0.0011Rd + 0.3449)V \quad (21)$$

$$R^2 = 0.995$$

La siguiente imagen (figura 60) muestra la superposición de las imágenes procesadas y tomadas con un desplazamiento radial de 2mm.

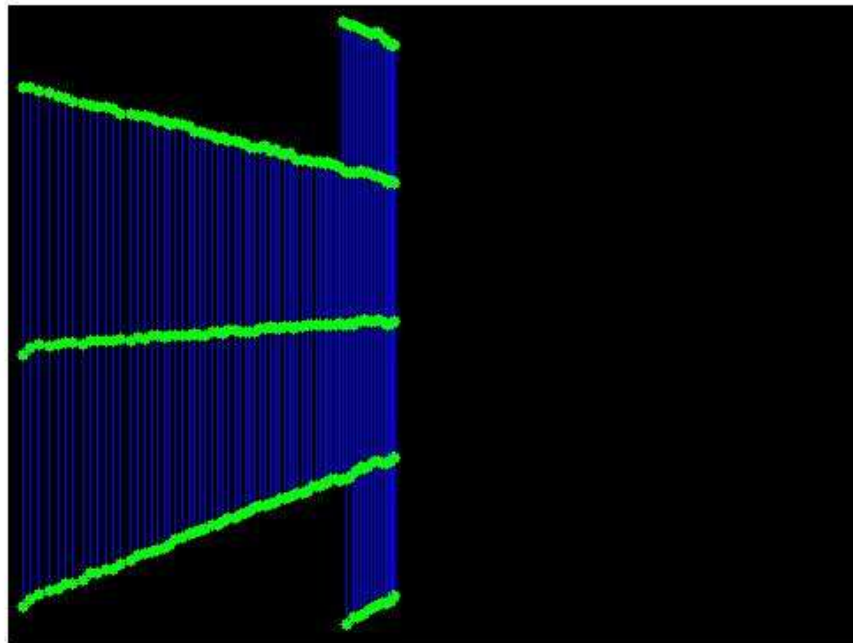


Fig. 60, Superposición de las imágenes con puntos de intersección.

* Aplicación de la curva de conversión en V.

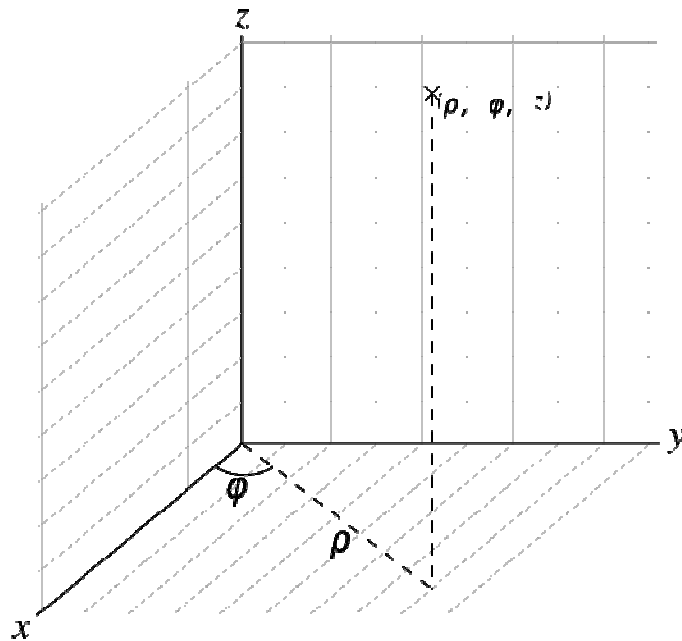


Fig. 61, Coordenadas cilíndricas

Una vez conocidas la distancia en milímetros ρ de cada punto de la línea láser deformada por la superficie del objeto con respecto al centro del sistema (Ecuación 17) y el ángulo φ que corresponde al desplazamiento angular del conjunto cámara-láser, el cual se encuentra en el extremo del eslabón que gira alrededor del objeto, se pasó a obtener la altura Z del punto (Ecuación 21), para así completar la terna de coordenadas cilíndricas que permiten ubicar un punto en el espacio (figura 61). A continuación se indica este proceso:

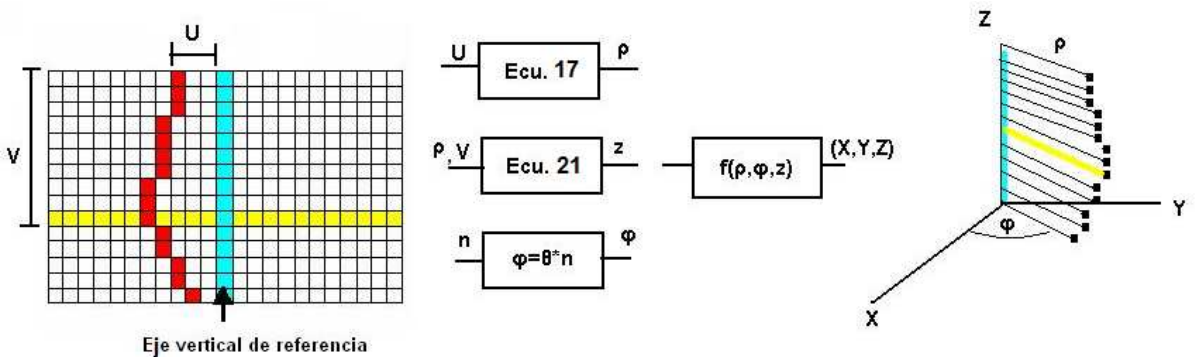
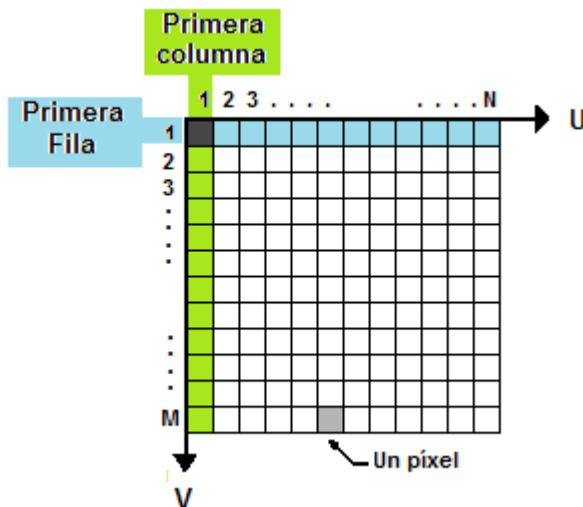


Fig. 62, Obtención de coordenadas cilíndricas y cartesianas.



La distancia en dirección U corresponde a la diferencia entre el eje vertical de referencia y el píxel correspondiente a la línea láser deformada en cada una de las filas.

La distancia en dirección V corresponde al número de la fila, siendo la primera de éstas la ubicada en el extremo superior de la imagen como se observa en la figura 63.

Fig. 63, Convención de coordenadas de las imágenes.

Para encontrar z en milímetros se necesita conocer ρ y la distancia en píxeles en dirección V .

Luego se convirtió las coordenadas cilíndricas (ρ, φ, z) a coordenadas cartesianas (X, Y, Z) con las siguientes ecuaciones:

$$X = \rho * \cos(\varphi); \quad (22)$$

$$Y = \rho * \sin(\varphi); \quad (23)$$

$$Z = z; \quad (24)$$

$$\varphi = \theta * n; \quad (25)$$

donde φ varía desde 0° hasta 360° en incrementos escogidos por el usuario (θ).

11. COLOR

Luego de la adquisición geométrica de la superficie del objeto que fue iluminada por el plano láser, se procede a adquirir el color de cada píxel que conforma la línea esqueletizada. Para éste escaneo se activa la luz blanca y permanece así hasta que el eslabón regrese al punto de partida.

Al finalizar, se genera un archivo .mat (archivo de Matlab) que contiene seis columnas con n filas, donde cada fila corresponde a un punto. Las tres primeras columnas corresponden a la información geométrica, es decir coordenadas espaciales compuestas por X, Y y Z. Las tres columnas restantes corresponden a los valores RGB de cada punto.

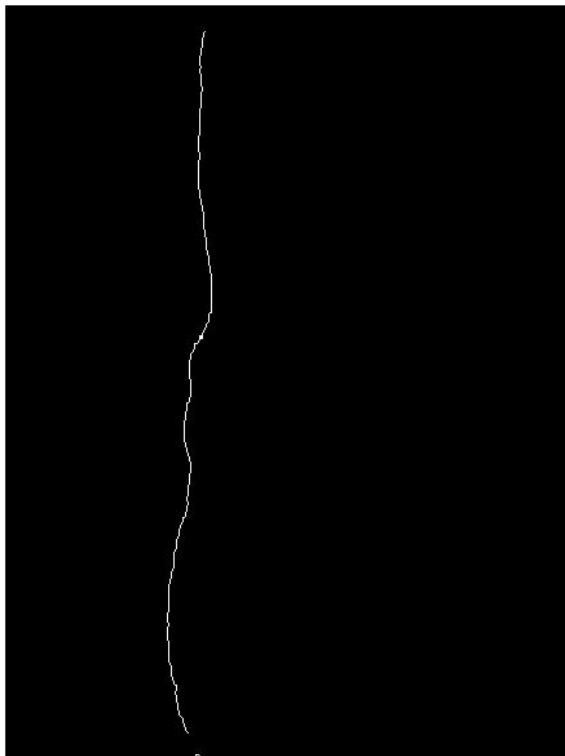


Fig. 64, Perfil de una pieza cerámica.

La figura 64, es el resultado del preprocesamiento y procesamiento de imágenes. Consiste en el plano láser proyectado en el objeto. El ancho de la línea es de un píxel.

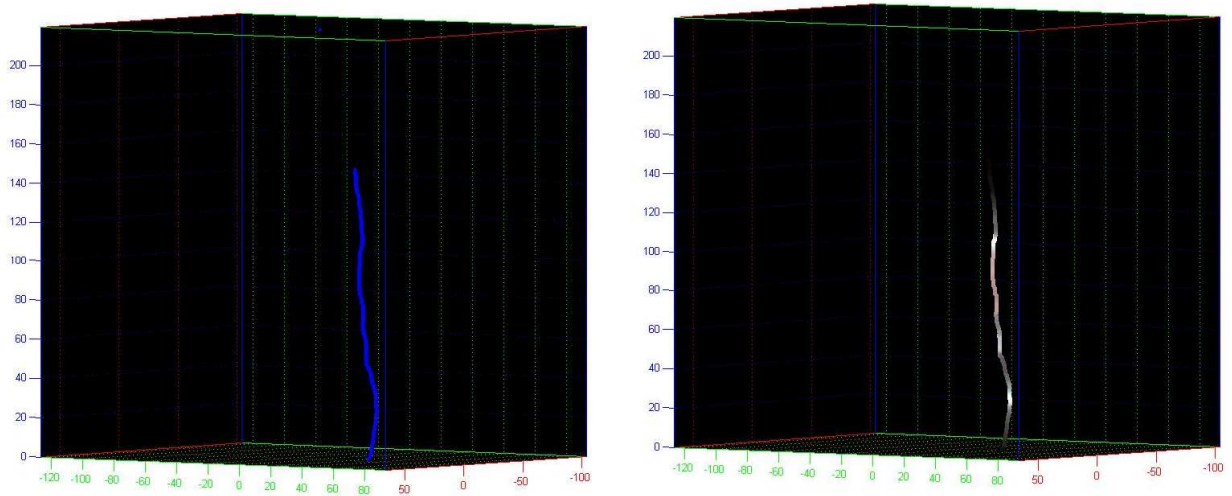
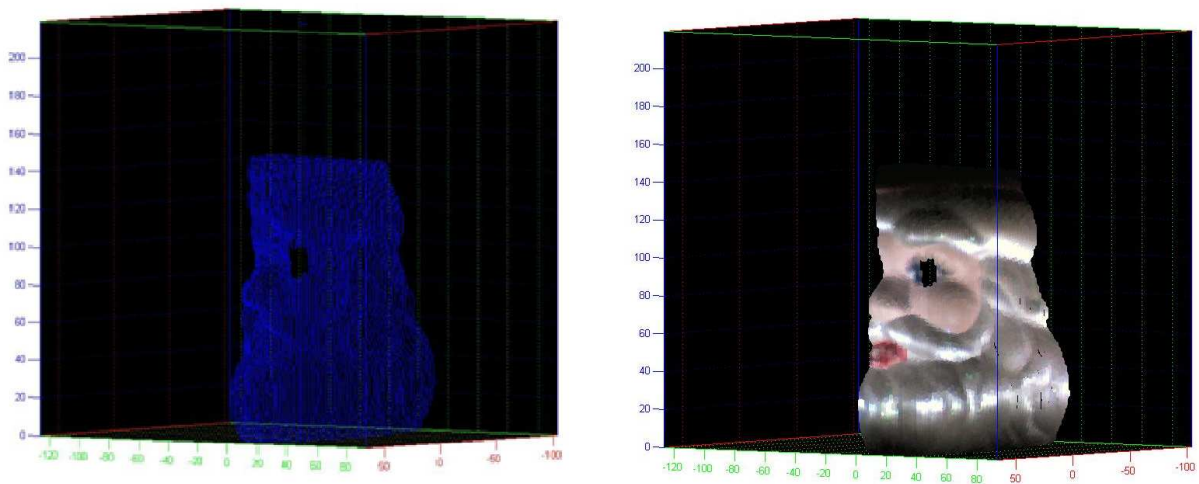


Fig. 65, Perfil; a. sin color, b. con color.

Con la posición de los píxeles que conforman la línea (u,v) y con las ecuaciones obtenidas en el proceso de calibración, se generan los puntos espaciales del perfil que ésta representa (figura 65.a). La figura 65.b muestra el mismo perfil pero con información cromática.



a. b.

Fig. 66, Modelo; a. sin color b. con color

La figura 66.a, muestra 75 perfiles sin información cromática que se ubican en el primer octante. La figura 66.b, muestra la misma información geométrica junto con su información cromática.

12. CARACTERIZACIÓN DEL SISTEMA

Se determinaron los parámetros que caracterizan las medidas obtenidas por el sistema como la resolución, la precisión, el rango de medidas y la densidad de la nube de puntos.

12.1 RESOLUCIÓN

La resolución es el incremento mínimo en la entrada para el que se obtiene un cambio en la salida.

12.1.1 Resolución en profundidad ρ . Como la profundidad, que en coordenadas polares corresponde al radio Rd , se mide por el desplazamiento en píxeles en dirección U y con respecto a la línea de referencia, entonces, el mínimo valor en X y Y será aquel producido por el desplazamiento de un píxel.

La resolución en la profundidad Rd se encontró reemplazando $U=0$ píxel y $U=1$ píxel en la curva de calibración de Rd y restándolos como se indica a continuación:

$$\delta Rd = Rd(1) - Rd(0) = 1.03 \text{ mm} \quad (26)$$

Como ρ es:

$$\rho = \frac{Rd}{\cos 15^\circ} \quad (27)$$

La resolución de ρ queda determinada como:

$$\delta \rho = \frac{\delta Rd}{\cos 15^\circ} = 1.06 \text{ mm} \quad (28)$$

12.1.2 Resolución en altura Z. Es la máxima separación en milímetros equivalente a una separación de un píxel en dirección del eje V. Se encontró reemplazando $Rd=0$ y $V=1$ en la ecuación de Z.

$$Z = f(U, V) = (2E - 7Rd^2 - 0.0011Rd + 0.3449)V \quad (29)$$

Dando como resultado:

$$\delta Z = 0.3449\text{mm} \quad (30)$$

12.2 PRECISIÓN EN EL DESPLAZAMIENTO ANGULAR

Para lograr que el sistema de desplazamiento angular incrementará θ grados, se envió una determinada cantidad de pulsos eléctricos al motor paso a paso principal que se encuentra acoplado al eslabón donde se encuentra la cámara y el láser por medio de cuatro engranes.

Un factor que afecta el desplazamiento angular es el rozamiento entre el acrílico y las ruedas que sostienen al eslabón. Como resultado de este fenómeno, el eslabón no da una vuelta completa.

El ángulo que falta para completar la vuelta es 1° . Si se distribuye 1° en forma uniforme sobre todo el trayecto, la desviación promedio en cada posición será de:

$$\delta_{prom\theta} = \frac{1^\circ}{\text{cantidad de perfiles}} = \frac{1^\circ}{n} \quad (31)$$

Para un incremento en el desplazamiento angular θ de 1.2° , que es el más usado, produce 300 perfiles y una desviación promedio de:

$$\delta_{prom\theta} = \frac{1^\circ}{300} = 0.00333 \dots^\circ \quad (32)$$

De tal forma que la precisión en la medida de φ está dada por:

$$\varphi = n\theta \pm \delta_{prom\theta} \quad (33)$$

12.3 RANGO

Las dimensiones máximas que debe tener un objeto para su reconstrucción 3D están limitadas por la distancia de observación de la cámara, que consiste en la distancia entre el centro de la base y la cámara.

Para una distancia de observación de 300mm, el rango de medida, es:

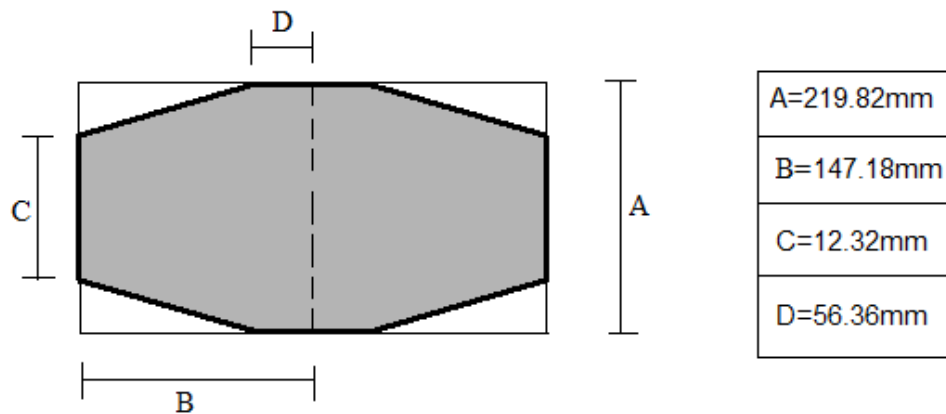


Fig. 67, Rango de medidas.

A: máxima distancia de medida en Z en el centro de la base.

Se encontró reemplazando $U=0$ (línea de referencia) en la ecuación de Profundidad Rd y éste valor se utilizó en la ecuación de Z con $V=640$ píxeles.

B: radio máximo (ρ max.) o profundidad máxima a partir del centro de la base que en coordenadas cartesianas corresponde al máximo valor que pueden tomar X y Y.

Se encontró reemplazando $U=224$ (máximo desplazamiento en U) en la ecuación de Profundidad Rd ($Rd=142.16\text{mm}$) y luego reemplazando éste valor en la ecuación de ρ .

C: máxima distancia de medida en Z a 142.16mm del centro de la base.

Se encontró reemplazando $Rd=142.16\text{mm}$ y $V=640$ píxeles en la ecuación de Z.

D: distancia desde el centro de la base hasta la coordenada con altura cero.

Se encontró midiendo el radio del círculo formado al escanear la base (sin ningún objeto).

12.4 DENSIDAD DE LA NUBE DE PUNTOS

La densidad de la nube de puntos depende del incremento del desplazamiento angular θ y de la altura del objeto. A continuación se indica la densidad máxima para diferentes valores de θ junto con el tiempo de escaneo aproximado, utilizando un computador Hewlett-Packard Pavilion dv6000 con procesador Intel® Core™2 Duo T5300 de 1.73 Ghz y Memoria RAM de 2Gb:

Incremento en el desplazamiento angular θ (grados)	Cantidad de perfiles	Cantidad máxima de puntos espaciales	Tiempo de escaneo (minutos)
0.6	600	384000	18
0.8	450	288000	15
0.9	400	256000	14
1.2	300	192000	12
1.8	200	128000	10
3.6	100	64000	8

Tabla 5, Densidad de la nube de puntos y tiempo de escaneo.

La cantidad de puntos solamente es máxima si se obtiene una coordenada para cada fila de píxeles de cada imagen, es decir 640 coordenadas por imagen. Sin embargo, este hecho casi no ocurre porque el tamaño del objeto hace que la proyección de la línea no se extienda a lo largo de toda la imagen o porque hay un agujero o una prolongación del objeto que no permite a la cámara capturar alguna parte de la línea.

Si el objeto a digitalizar es pequeño, la línea deformada será de longitud menor que la obtenida con un objeto grande. Por tal motivo las coordenadas de los píxeles correspondientes a la línea serán menos, lo que nos lleva a concluir que con objetos pequeños la densidad en la nube de puntos es menor.

12.5 CARACTERÍSTICAS ELÉCTRICAS

El digitalizador tridimensional utiliza una fuente de computador la cual suministra 12v. La corriente máxima que consume en funcionamiento normal es 420mA. El consumo de corriente depende de los elementos que estén en funcionamiento como se indica en la tabla 6.

Motor de giro	Motor de calibración	Luz blanca	Láser	Microcontrolador y demás CI.	Corriente (mA)
				X	110
X			X	X	240
X		X		X	420
	X		X	X	220

Tabla 6, Consumo de Corriente

13. VALIDACION

El proceso de validación se realizó con el fin de comparar valores ideales con los valores medidos por el sistema de adquisición de geometría tridimensional y analizar que tan cercanos están (exactitud).

La digitalización se realizó a tres objetos; un cilindro, un prisma rectangular y una esfera. El incremento en el desplazamiento angular fue de 1.8° , resultando 200 perfiles. Las gráficas que se muestran a continuación se realizaron en Matlab.

13.1 CILINDRO

Para un cilindro de 93 milímetros de radio y 186 milímetros de altura, el cual se muestra en la figura 68, se obtuvo el modelo tridimensional indicado en la figura 69.



Fig. 68, Objeto a digitalizar

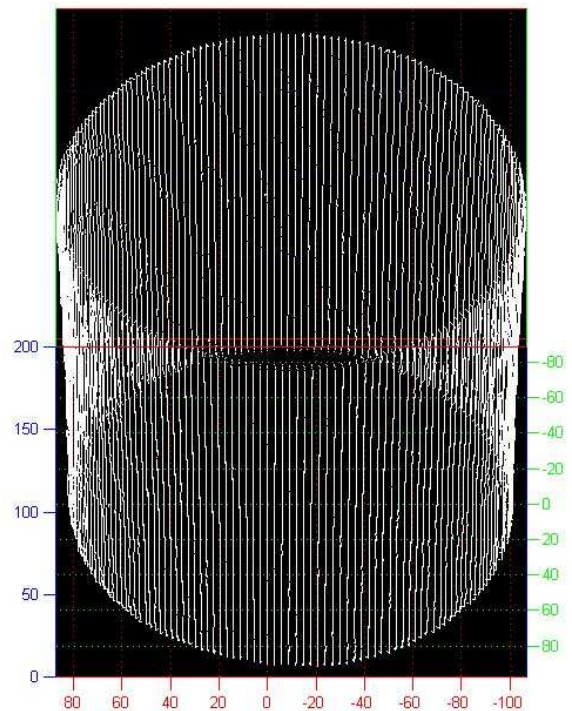


Fig. 69, Modelo tridimensional

Se realizó un corte transversal a una altura de 140 mm y se graficó en un plano todos los puntos que se encuentran dentro de un intervalo de 10 mm por encima de esta altura, es decir hasta 150 mm, obteniéndose el círculo que se indica a continuación:

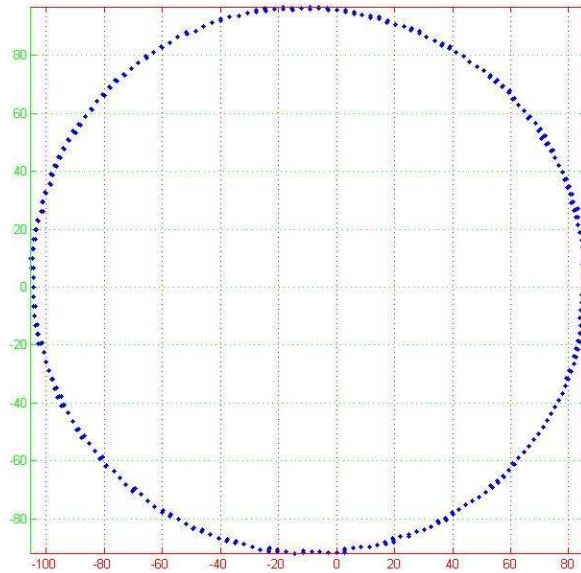


Fig. 70, Corte transversal del objeto digitalizado

Luego se comparó gráficamente los datos escaneados con datos ideales que representan un círculo de radio 93 mm y con centro en (0,0). Las figuras 71 y 72 muestran esta situación:

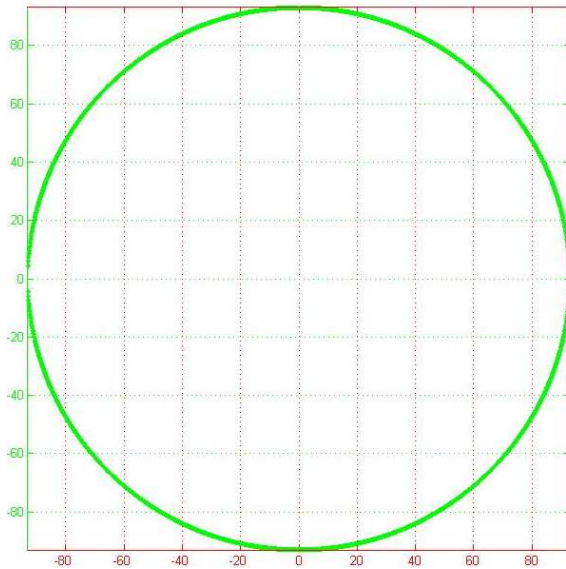


Fig. 71, Círculo ideal

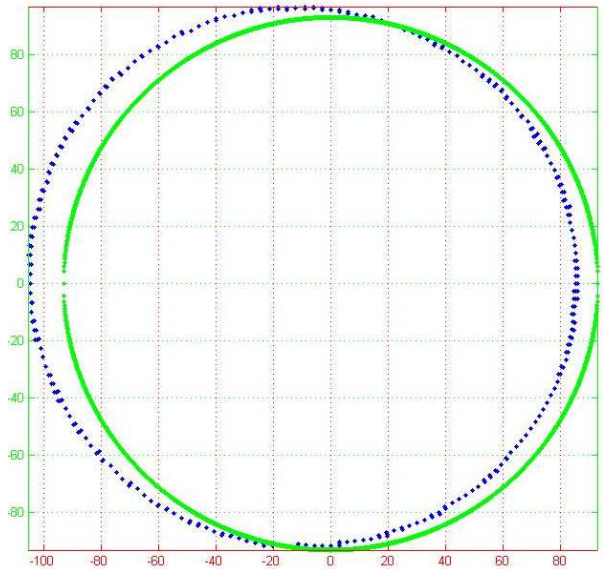


Fig. 72, Comparación

Se cambió el centro del círculo ideal a $(-10,3)$ que corresponde aproximadamente al centro del círculo escaneado. La comparación se indica en la figura 73.

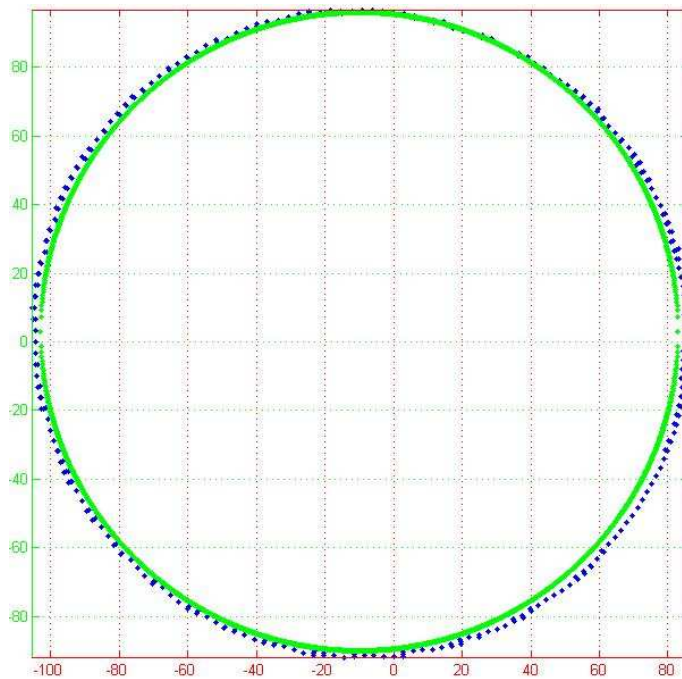


Fig. 73, Círculo ideal y escaneado concéntricos

Se realizó la misma prueba pero ubicando el cilindro de tal forma que el centro del círculo correspondiente a su corte transversal quede mucho más alejado del centro del sistema.

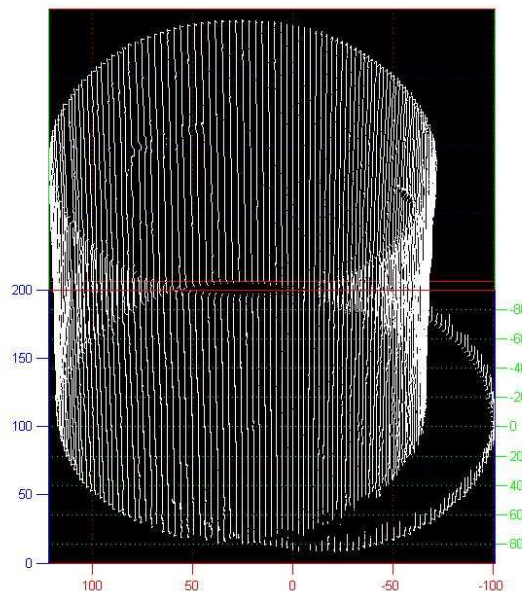


Fig. 74, Cilindro digitalizado con centro en $(26,-4)$

El corte transversal, la comparación original y la comparación luego de cambiar el centro del círculo ideal, se muestra a continuación:

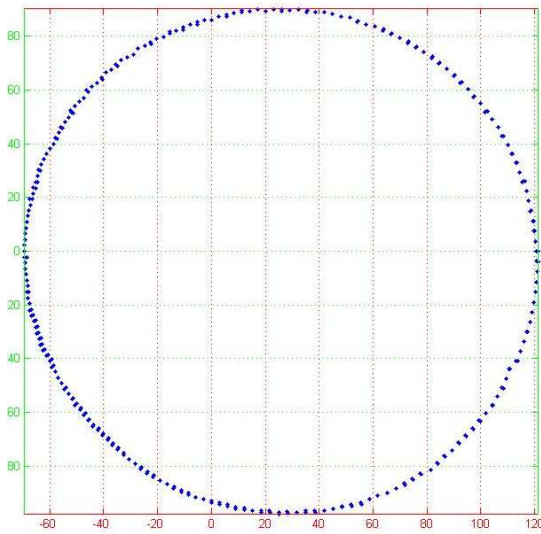


Fig. 75, Corte transversal

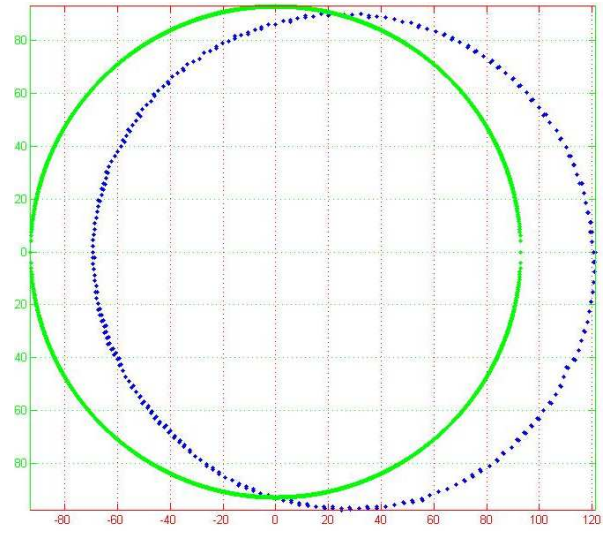


Fig. 76, Comparación

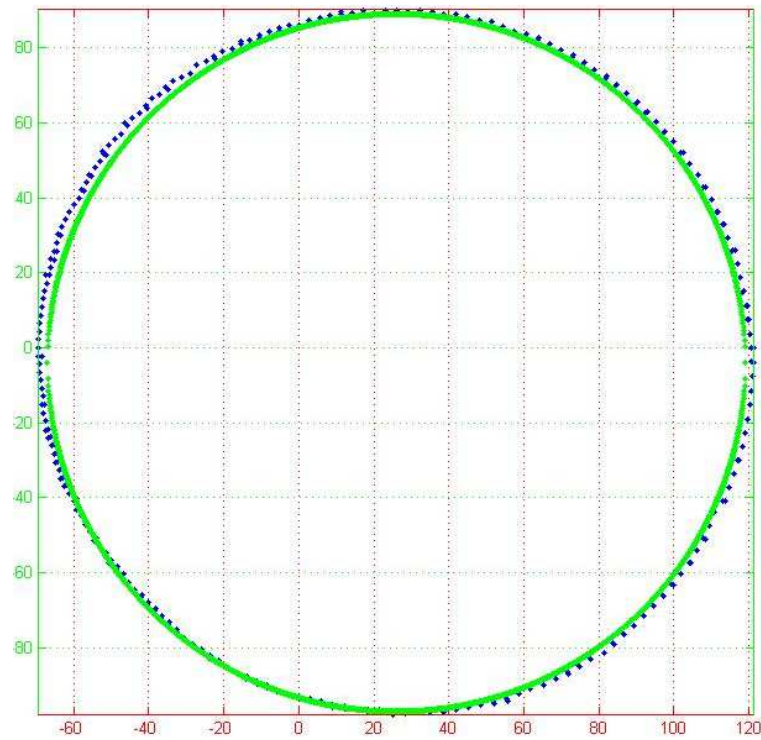


Fig. 77, Círculo ideal y escaneado concéntricos

13.2 PRISMA RECTANGULAR

Se desea conocer el comportamiento del sistema para objetos que tienen cambios bruscos en su superficie como es el caso de las esquinas de un prisma rectangular indicado a continuación:

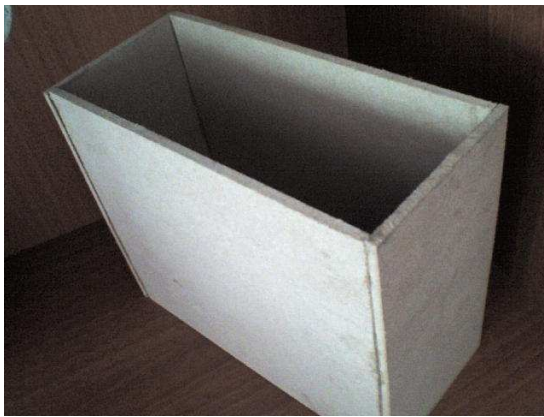


Fig. 78, Objeto a digitalizar

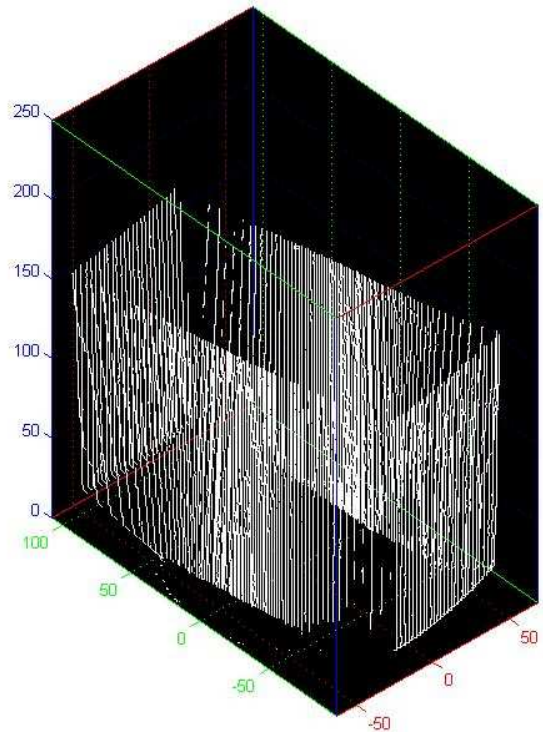


Fig. 79, Modelo tridimensional

Las dimensiones del prisma rectangular son: 190 mm de largo, 85 mm de ancho y 160 mm de alto.

Se realizó un corte transversal a una altura de 140 mm y se graficó en un plano todos los puntos que se encuentran dentro de un intervalo de 10 mm por encima de esta altura, es decir hasta 150 mm, obteniéndose el rectángulo que se indica a continuación:

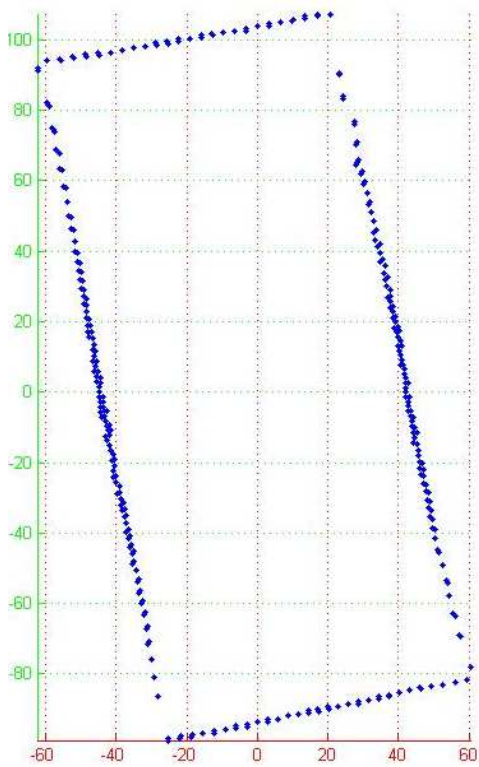


Fig. 80, Corte transversal



Fig. 81, Rectángulo ideal

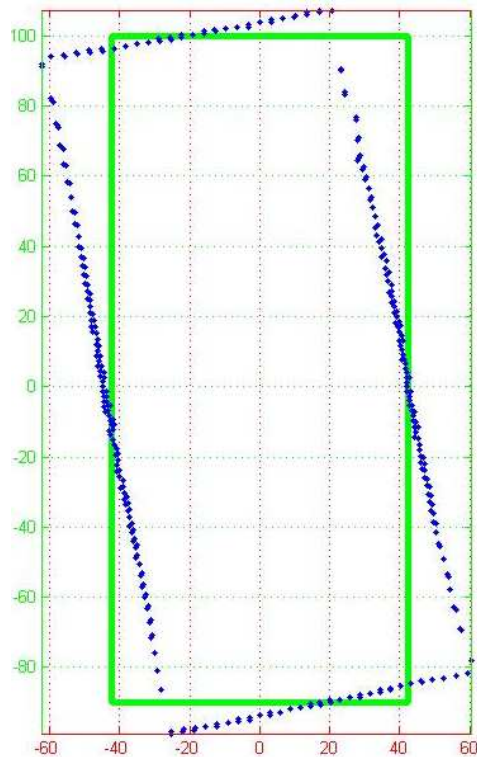


Fig. 82, Comparación

No se puede hacer una adecuada comparación gráfica debido a que no coinciden las orientaciones de los rectángulos (figura 82). Para compararlo con el rectángulo ideal de 190 mm de largo y 85 mm de ancho (figura 81), se necesita hacer una rotación de -11° con respecto a (0,0) como se indica a continuación:

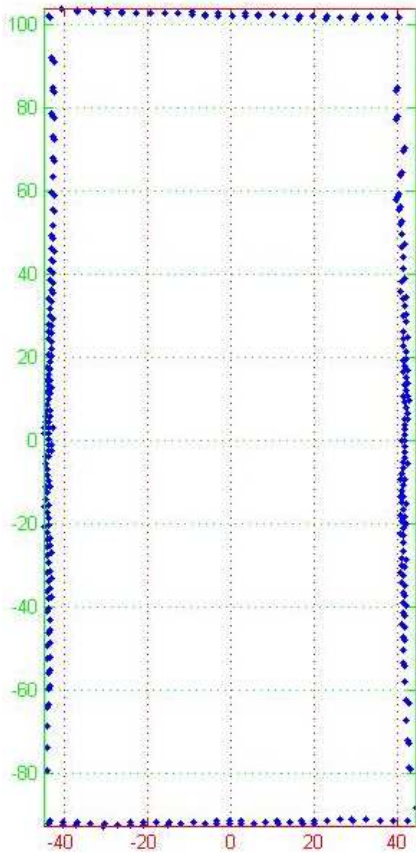


Fig. 83, Datos rotados -11°

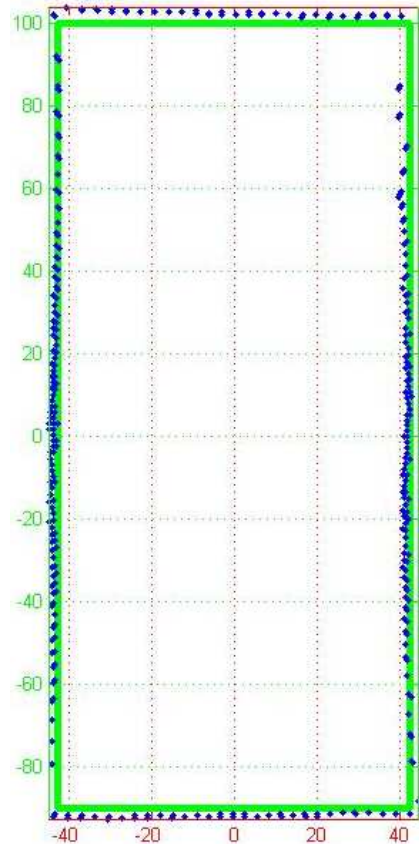


Fig. 84, Comparación

13.3 ESFERA

Luego de haber comprobado la exactitud de los datos de profundidad, se validaron los datos correspondientes a la altura. Para este fin, se digitalizó una esfera de icopor de 60 mm de radio, la cual se muestra a continuación junto con su modelo tridimensional:

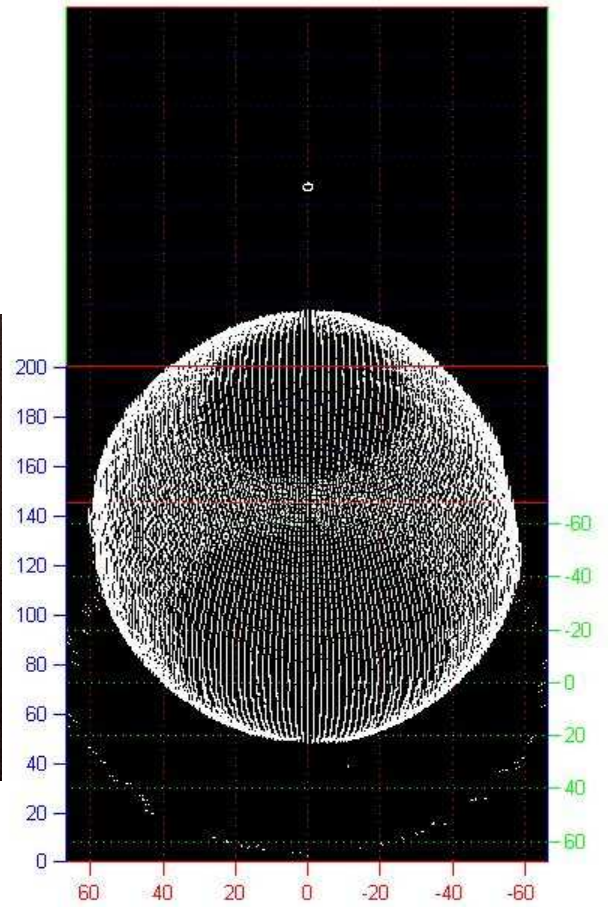


Fig. 85 Objeto a digitalizar

Fig. 86, Modelo tridimensional

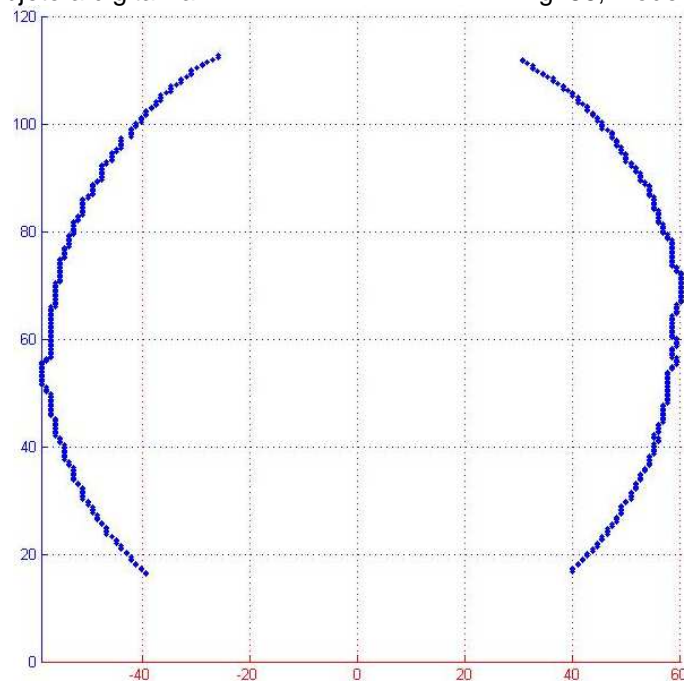


Fig. 87, Corte longitudinal

La anterior figura muestra el corte longitudinal de la esfera, correspondiente a la unión del perfil número 100 y 200, los cuales están separados 180°. No se observa la parte superior ni inferior de la circunferencia porque la cámara y/o el láser no llegan a estos lugares.

Luego se realizó la comparación con un círculo ideal, con centro en (0,0) y radio de 60 mm. El resultado se indica en la figura 88.

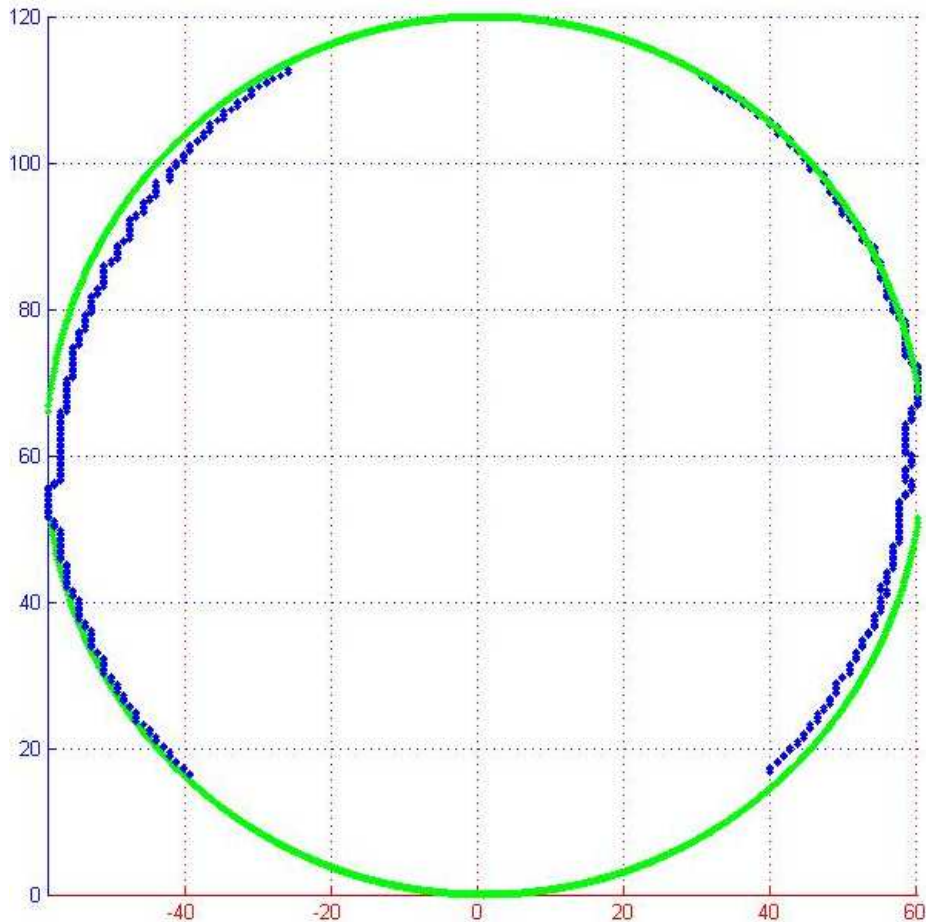


Fig. 88, Comparación

Se observa que hay una mayor desviación en el segundo y cuarto cuadrante (tomando como origen (0,60)), que corresponde a las imperfecciones de la esfera, debido a que está compuesta por dos semiesferas unidas por sus bases circulares, las cuales no son concéntricas. Esta situación se puede ver en la figura 85.

13.4 CÁLCULO DEL ERROR

Se tomaron 200 coordenadas del cilindro de la figura 69 y se trasladaron (10,-3) para que su centro quede en el origen. De esta manera, con sus coordenadas X y Y, se puede obtener el radio y compararlo con el verdadero valor que corresponde a 93 mm.

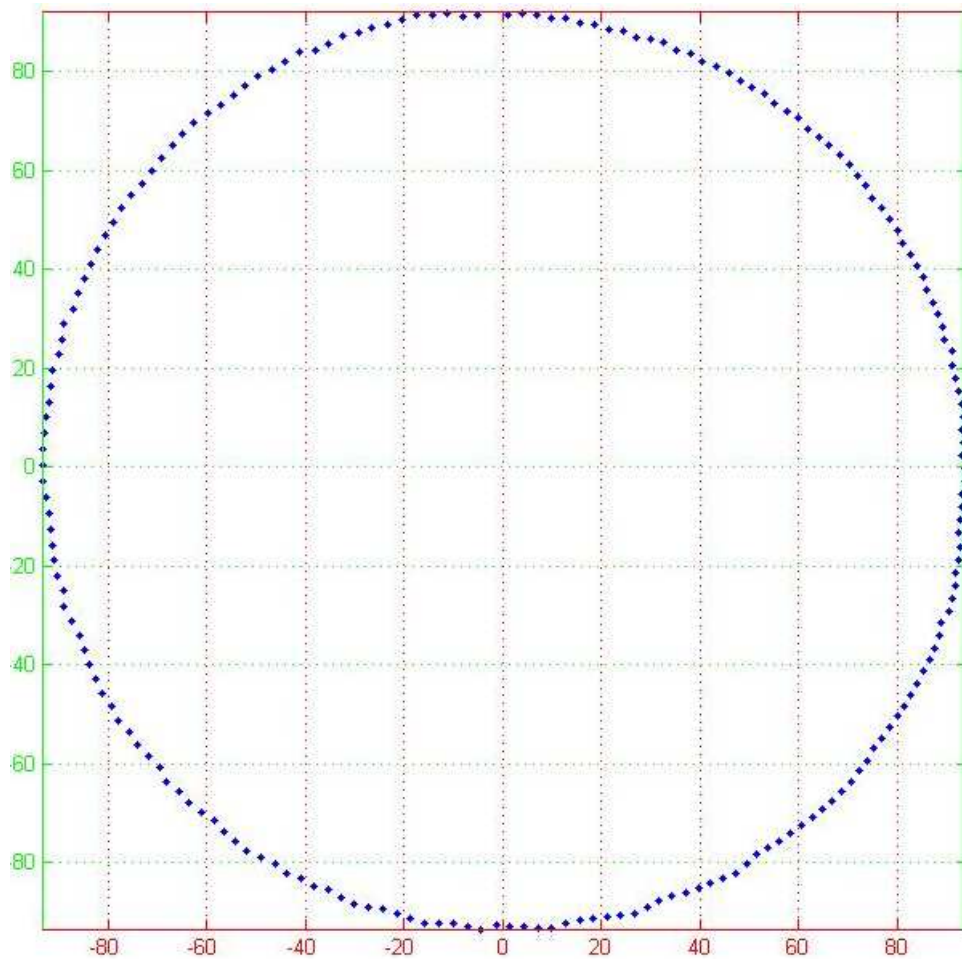


Fig. 89, Datos con centro en el origen

La determinación del error del instrumento permite conocer su grado de exactitud. Éste error fue evaluado mediante la siguiente ecuación:

$$e = \sqrt{\frac{1}{T} \sum (V_m - V_v)^2} \quad (34)$$

donde T es el número de datos, que para éste caso es 200.

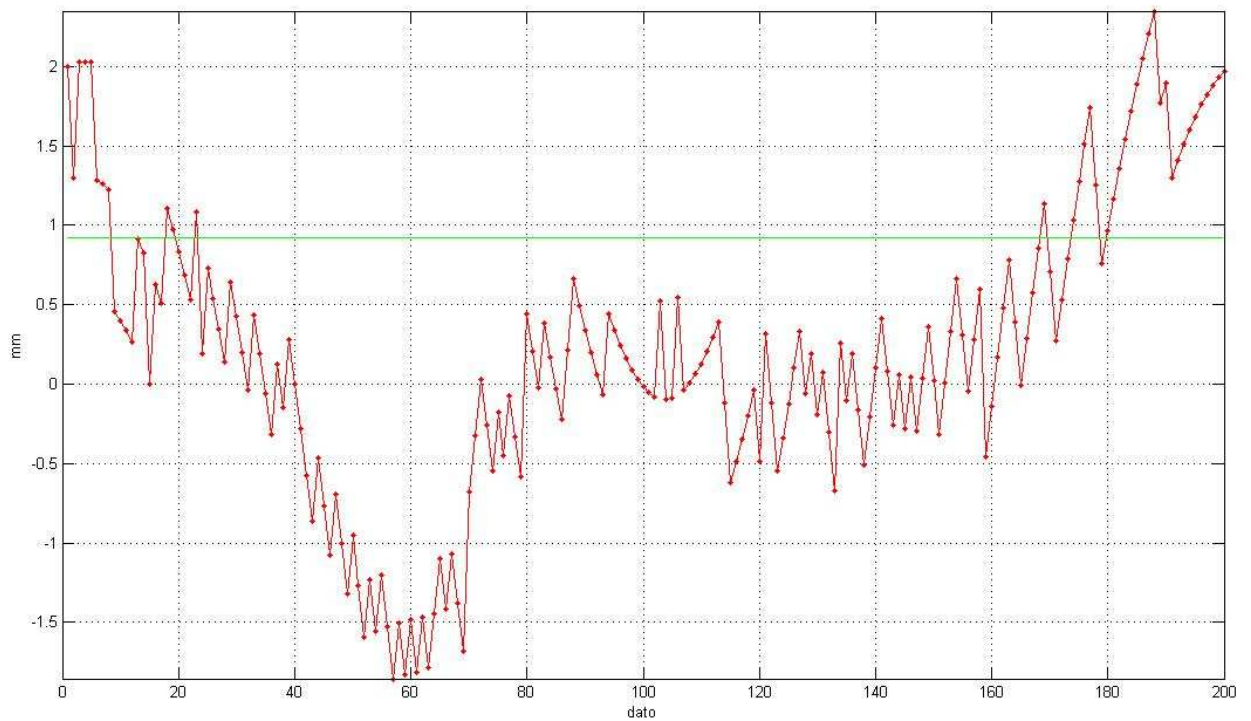


Fig. 90, Error del instrumento de medición

La figura 90 indica un error del instrumento de medición de 0.9217 mm (en verde), que representa un error relativo de 0.62% para mediciones de profundidad R_d dentro de un rango de 0 a 147.18 mm.

Además se indica el error absoluto para cada dato (en rojo), definiéndose éste como:

$$e_a = V_{medido} - V_{verdadero} \quad (35)$$

El radio medido con mayor magnitud se encuentra a 2.3489 mm por encima del radio verdadero.

El radio medido con menor magnitud se encuentra a 1.8553 mm por debajo del radio verdadero.

14. RESULTADOS



Fig. 91, Fotografía de un Santa Claus

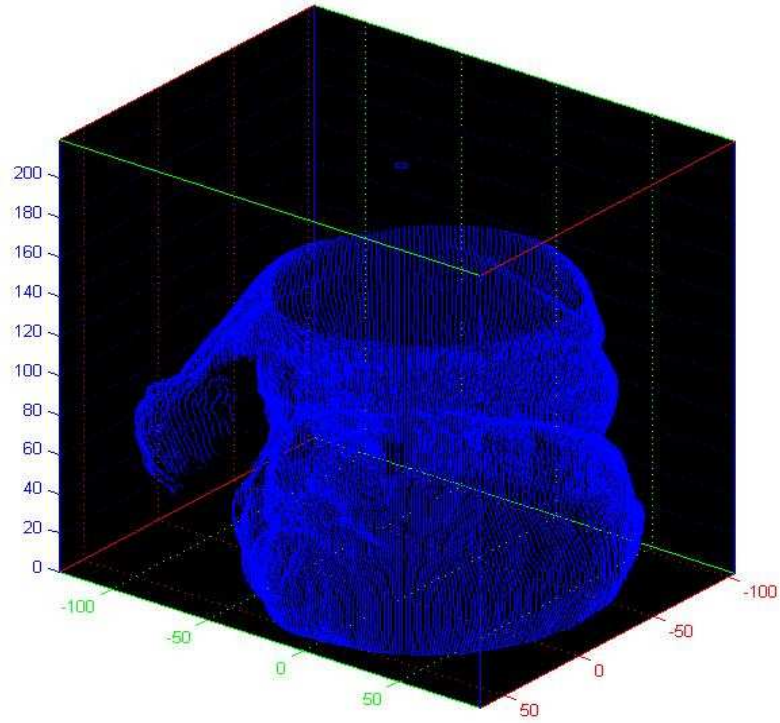


Fig. 92, Nube de puntos en Matlab.

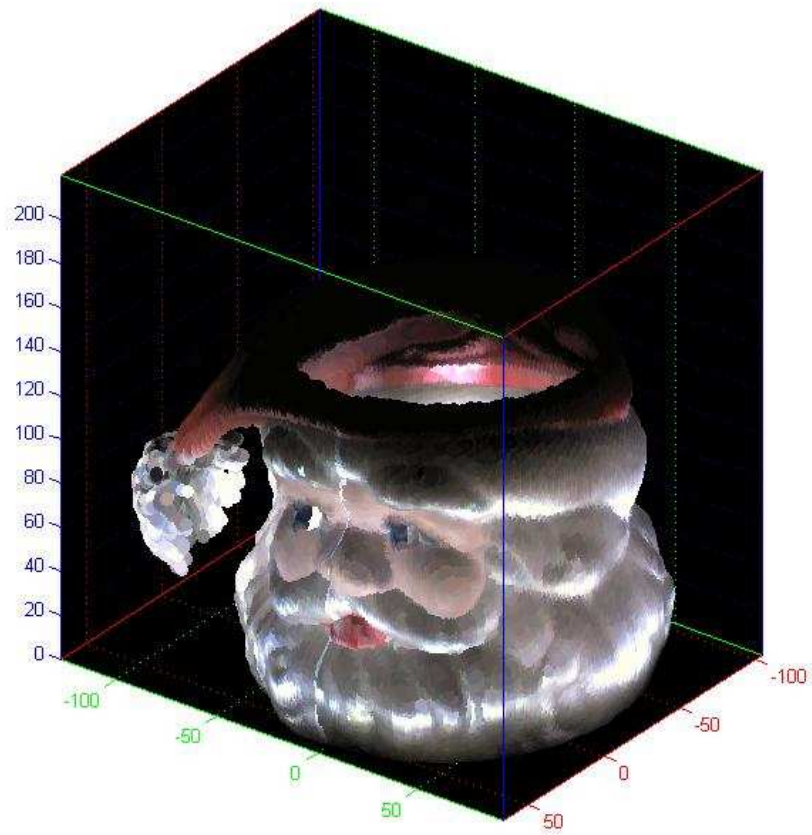


Fig. 93, Modelo con información cromática en Matlab.

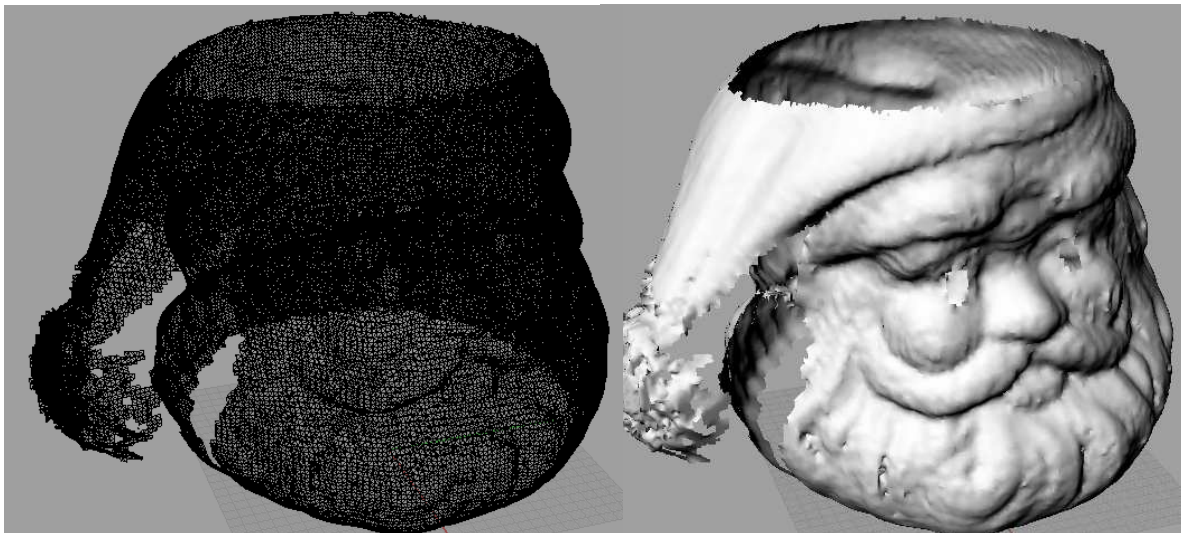


Fig. 94, Estructura alámbrica y superficie (renderizada) en Rhinoceros



Fig. 95, Fotografía de un cráneo humano.

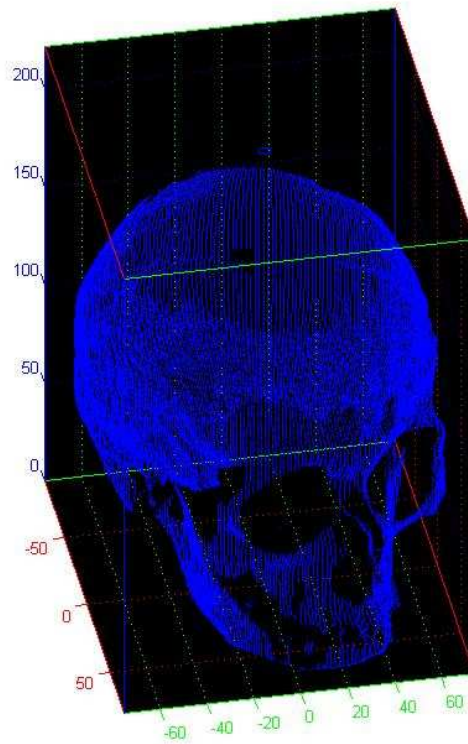


Fig. 96, Nube de puntos en Matlab.

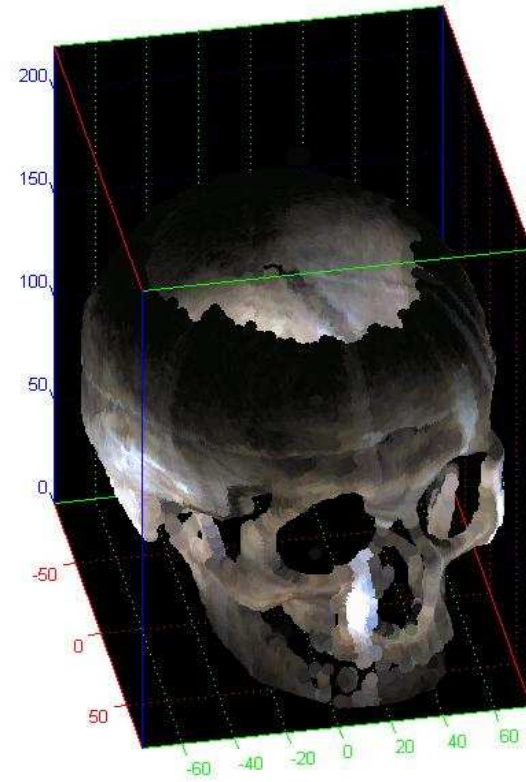


Fig. 97, Modelo con información cromática en Matlab.

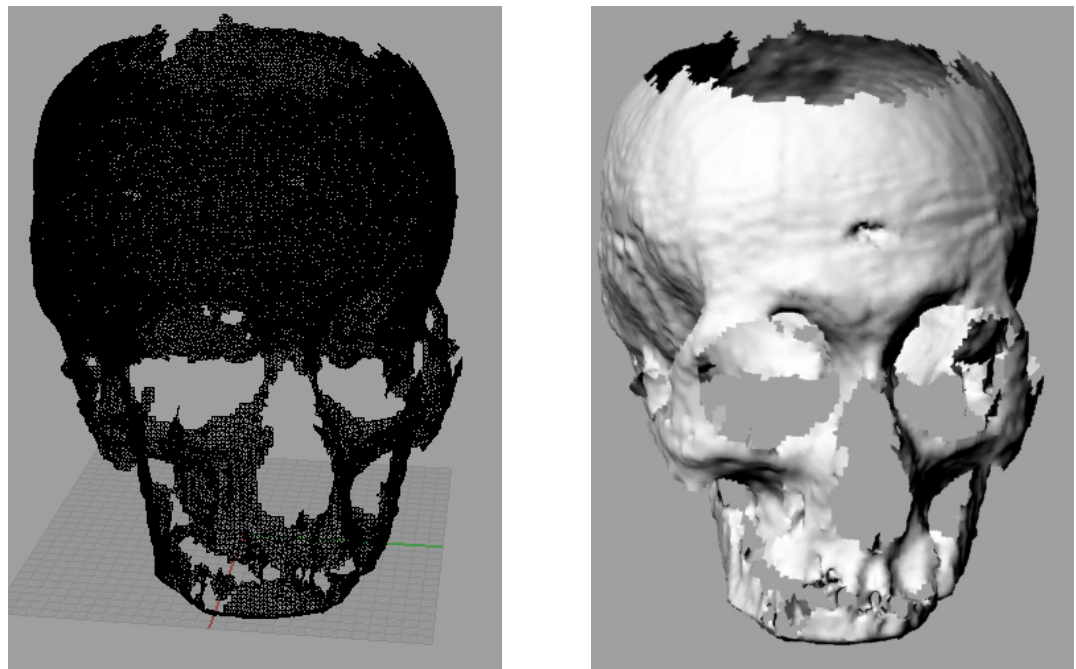


Fig. 98, Estructura alámbrica y superficie (renderizada) en Rhinoceros.



Fig. 99, Fotografía de Garfield.

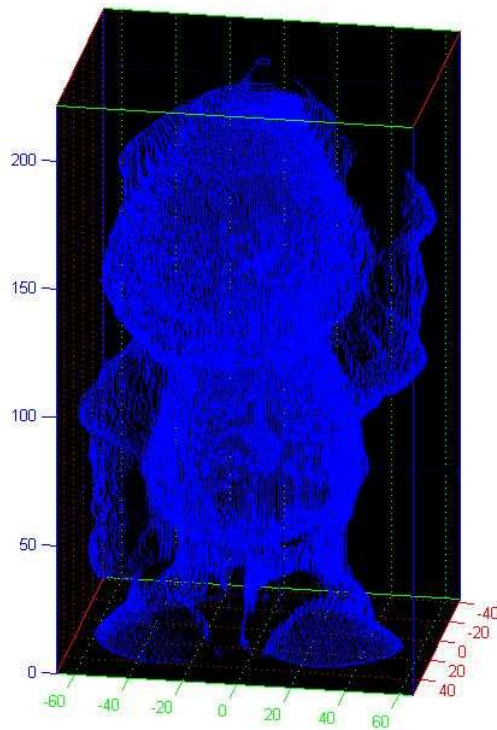


Fig. 100, Nube de puntos en Matlab.

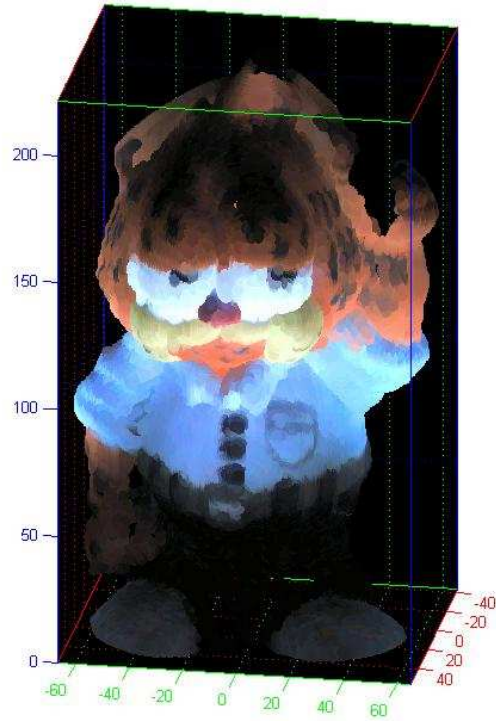


Fig. 101, Modelo con información cromática en Matlab.

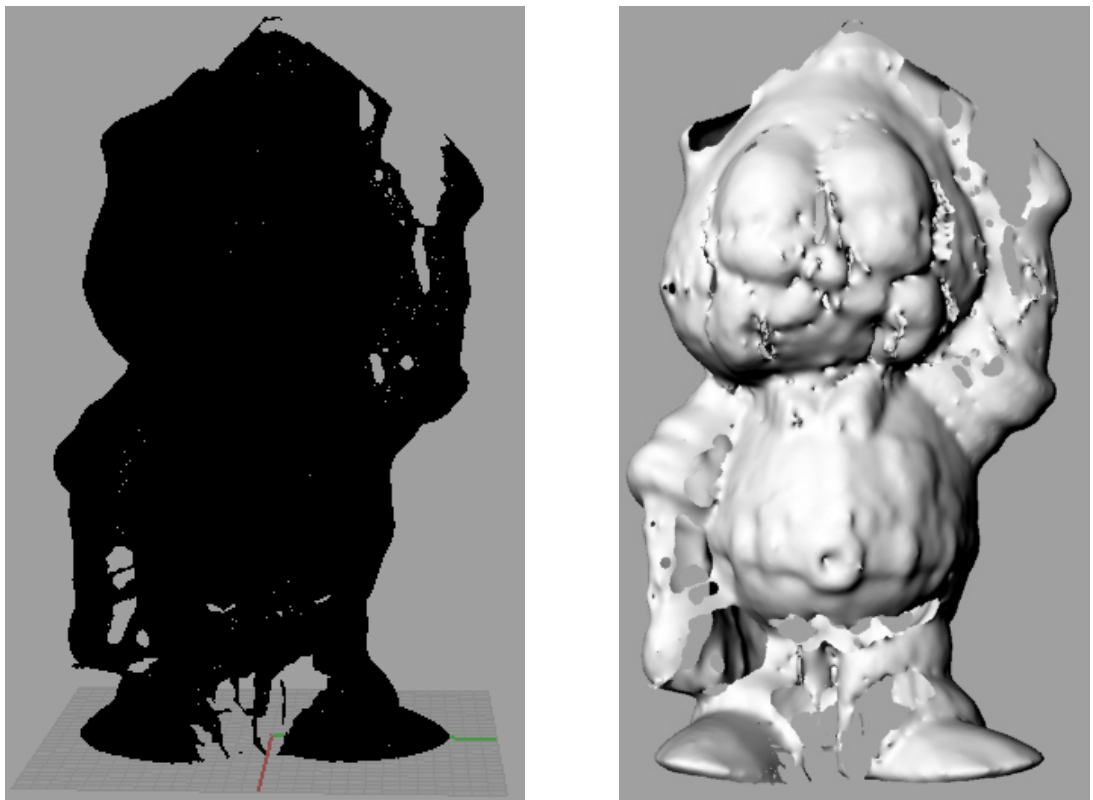


Fig. 102, Estructura alámbrica y superficie (renderizada) en Rhinoceros.



Fig. 103, Fotografía de un faraón.

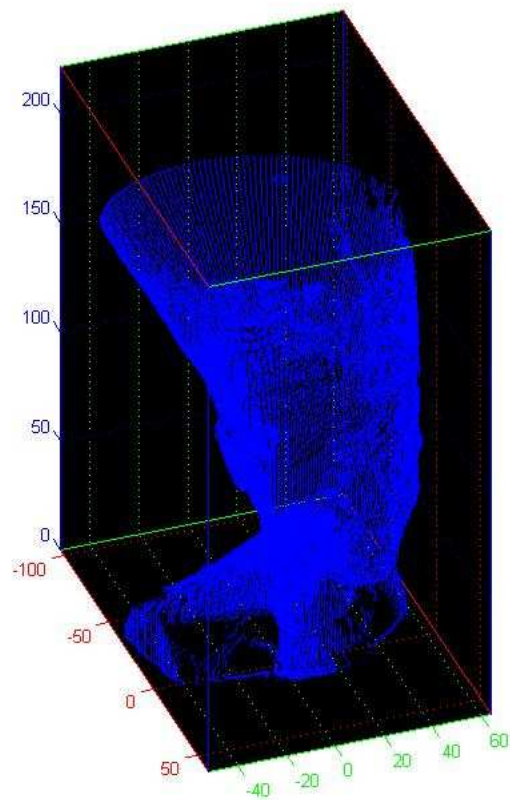


Fig. 104, Nube de puntos en Matlab.

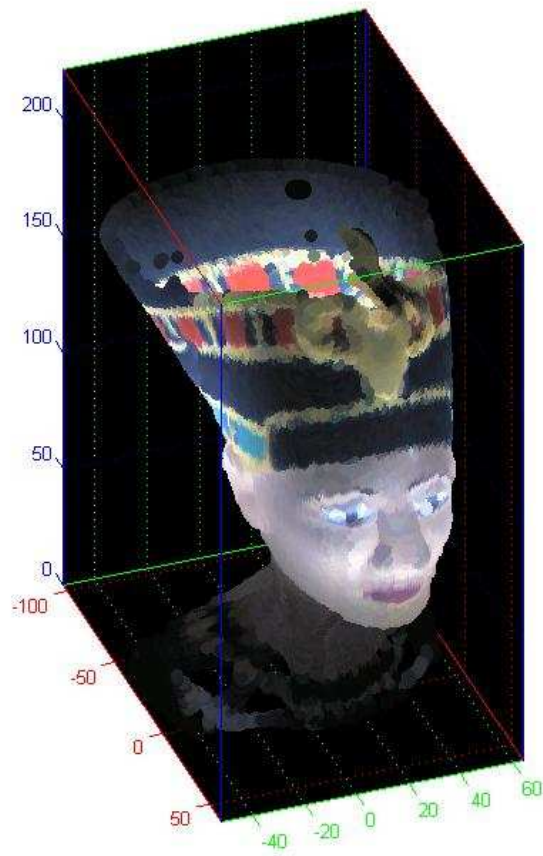


Fig. 105, Modelo con información cromática en Matlab.

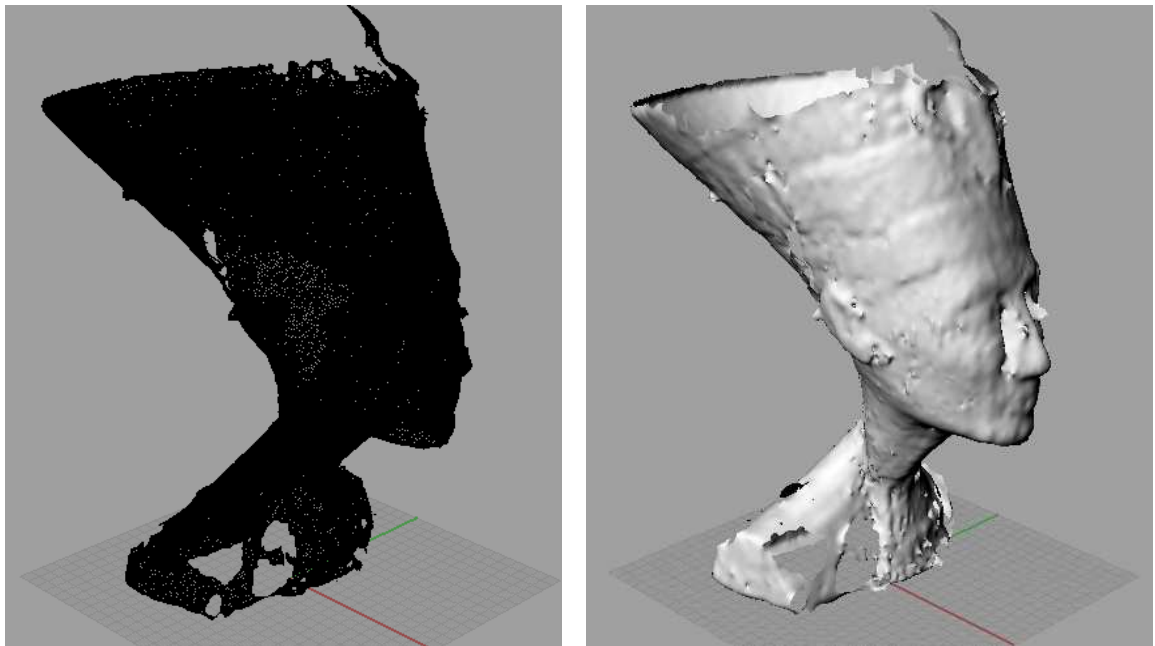


Fig. 106, Estructura alámbrica y superficie (renderizada) en Rhinoceros.



Fig. 107, Fotografía de un bebé.

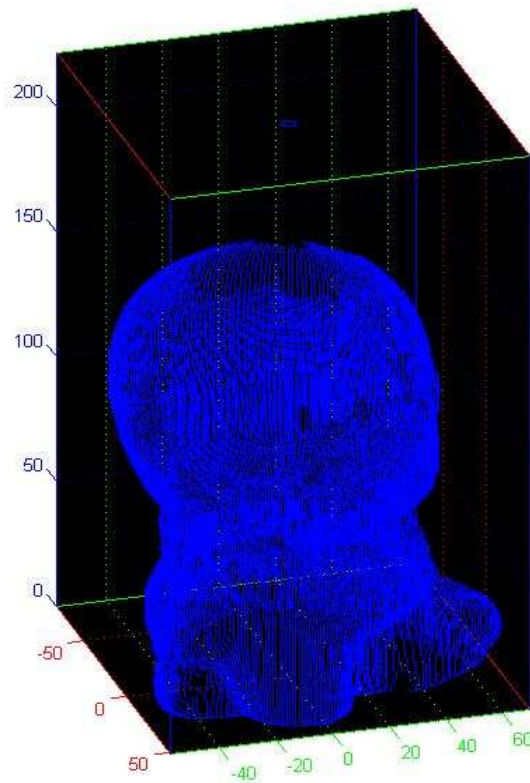


Fig. 108, Nube de puntos en Matlab.

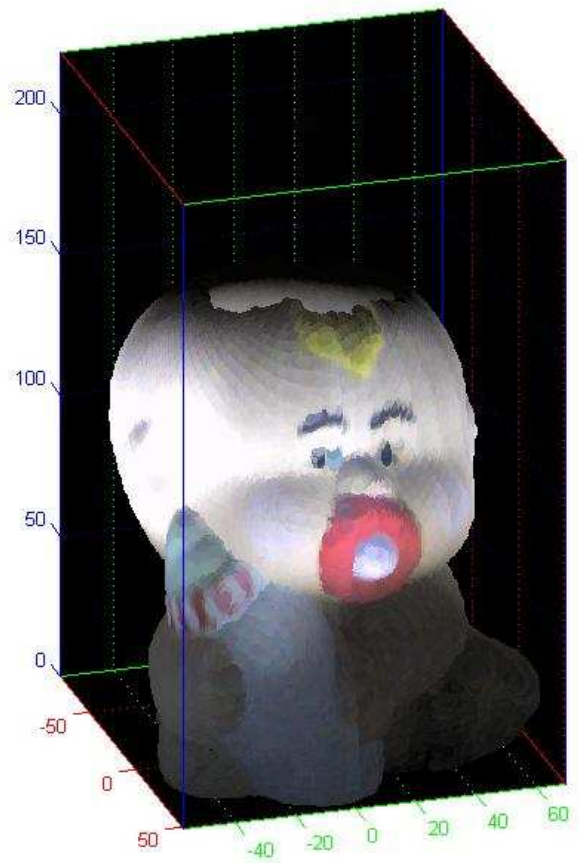


Fig. 109, Modelo con información cromática en Matlab.

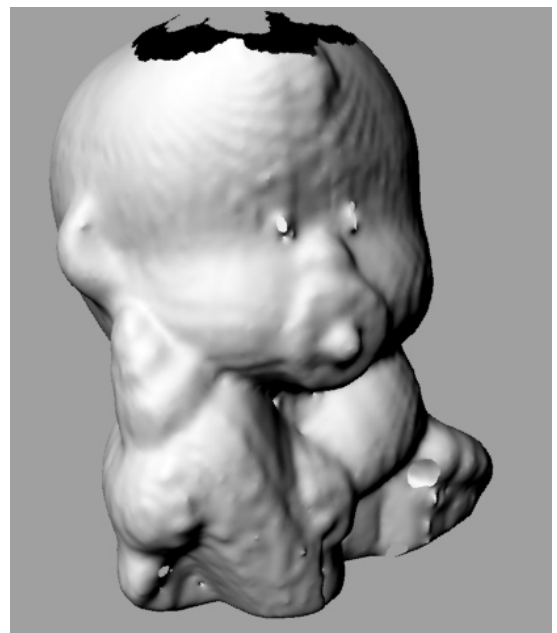
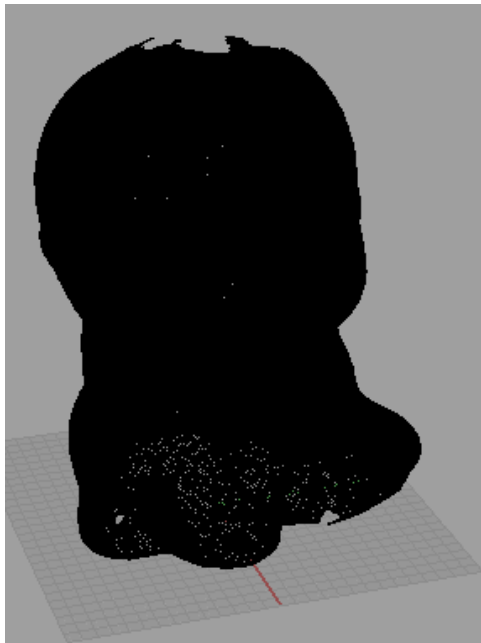


Fig. 110, Estructura alámbrica y superficie (renderizada) en Rhinoceros.

14.1 SOFTWARE DEL DIGITALIZADOR

Son los programas diseñados para la obtención de datos, análisis de los mismos y calibración del digitalizador. Se desarrolló con el toolbox de interfaces gráficas que posee MATLAB®.

Se optó por desarrollarlos en MATLAB® por su robustez en el análisis matemático y la simplicidad que lo caracteriza para el procesamiento de imágenes y gestión de datos.

El software del digitalizador consta de tres programas:

14.1.1 Software de captura.

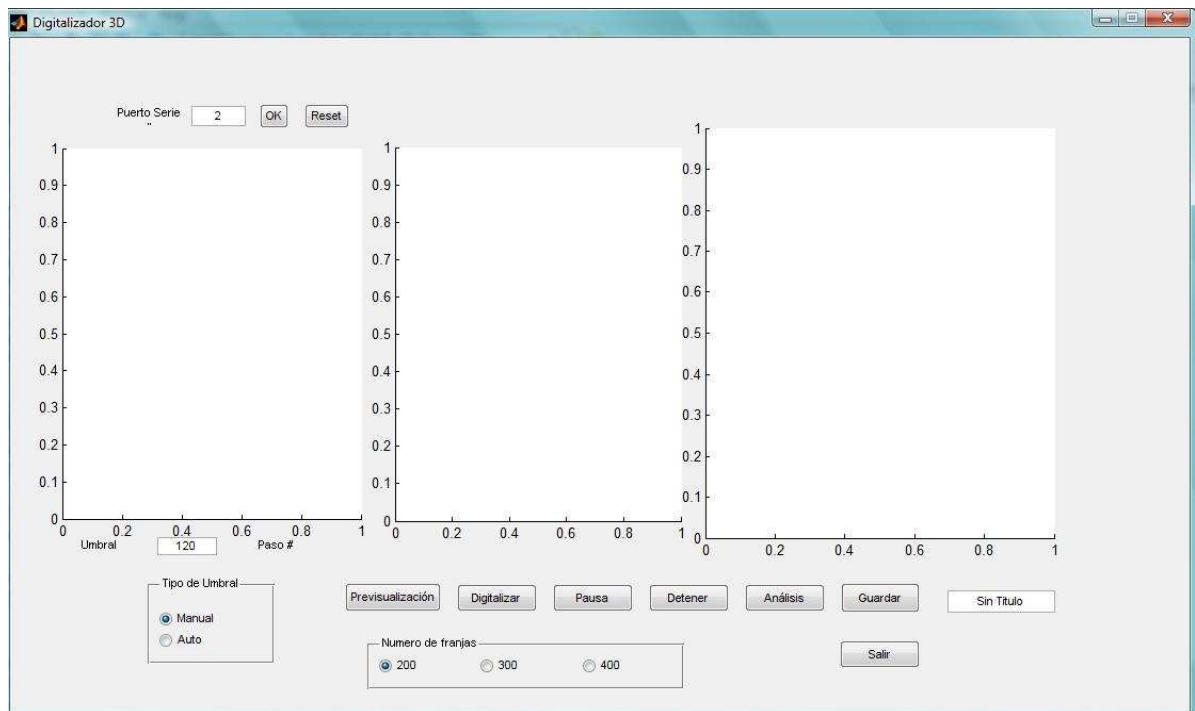


Fig. 111, Software de captura.

Este programa es el encargado de crear una nube de puntos a partir del procesamiento de las fotografías del objeto y se divide en tres etapas.

* Etapa de pre visualización del análisis de imágenes. Muestra al usuario el resultado del pre-procesamiento y procesamiento de imágenes utilizando el nivel del umbral que se selecciona manual o automáticamente de acuerdo al tipo, textura o color del objeto a digitalizar, permitiendo obtener datos más precisos.

El proceso consiste en especificar un umbral que puede ser digitado por el usuario o puede ser calculado por el computador. Luego permite visualizar varias etapas del análisis de imágenes para que el usuario determine cuál es el valor del umbral con mejores resultados (umbral manual). Entre esas visualizaciones se encuentra la imagen del objeto y la proyección del plano láser sobre éste, la imagen umbralizada y la imagen esqueletizada que generará las coordenadas espaciales del objeto.

En umbral automático, éste valor siempre varía de acuerdo con la cromaticidad de la zona del objeto que interseca el plano láser.

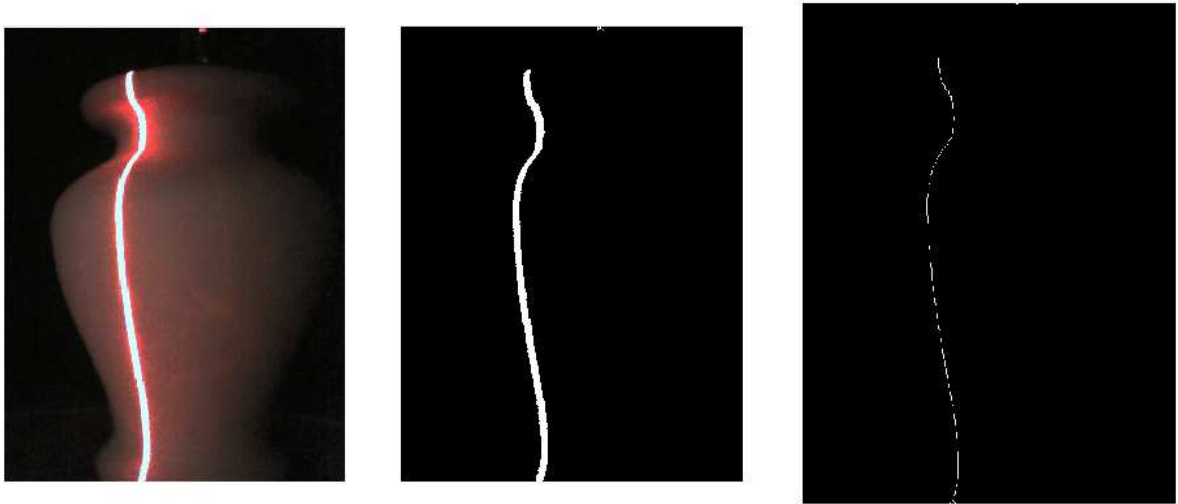


Fig. 112, Etapa de pre-visualización.

* Etapa de adquisición de nube de puntos. Esta etapa pone en funcionamiento toda la parte electro-mecánica del digitalizador, y se encarga de guardar en matrices la información geométrica y cromática que se obtiene para cada proceso de escaneo.

Comienza con la elección del puerto donde está conectado el circuito de control del digitalizador, la densidad de la nube de puntos y el valor del umbral.

La densidad de la nube de puntos se encuentra predeterminada a ciertos valores que dependen de la resolución máxima que posee el motor paso a paso y del arreglo de engranajes que mueven al eslabón. Las tres opciones de resolución son 200 300 y 360 franjas, que corresponden a un desplazamiento angular de 1.8° , 1.2° y 1° , respectivamente.

Una vez determinadas estas tres variables se da comienzo al proceso de obtención de las coordenadas de los píxeles que superan el umbral de intensidad. Primero se debe presionar el botón “Digitalizar” el cual generará las siguientes órdenes:

- Activación del láser
- Captura de imagen del objeto con el haz de luz del láser
- Pre-procesamiento y procesamiento de la imagen, borrando lo que no pertenezca al haz de luz.
- Procesamiento matemático que obtiene las coordenadas de los píxeles del haz de luz láser.
- Envío de datos al motor paso a paso para que el eslabón se mueva cierta cantidad de grados

Este proceso se repite una y otra vez hasta que la articulación cubre los 360°. Inmediatamente terminado este proceso el eslabón comenzará su recorrido en sentido contrario y se activará la iluminación interna del digitalizador con el fin de comenzar el proceso de obtención de color.

La opción “Pausa” se encarga de detener el proceso de captura de puntos hasta que el usuario decida proseguir de nuevo.

La opción “Detener” se encarga de terminar el proceso e inmediatamente envía la articulación a su posición inicial.

* Etapa de Análisis y Almacenamiento de datos. Esta etapa solamente puede ser activada después de que el proceso de digitalización se haya completado (se activa el botón de análisis).

Se encarga de transformar las coordenadas de posiciones dadas en píxeles y en el plano a coordenadas espaciales de acuerdo a las curvas de calibración en U y en V.

Una vez terminado el proceso de análisis se visualiza la nube de puntos y activa la opción “Guardar” que permite guardar las matrices de nubes de puntos en formato MAT para análisis en Matlab y en formato TXT para su análisis con programas dedicados al modelado 3d en NURBS como Rhinoceros 3D® y VR Mesh®.

14.1.2 Software de calibración.

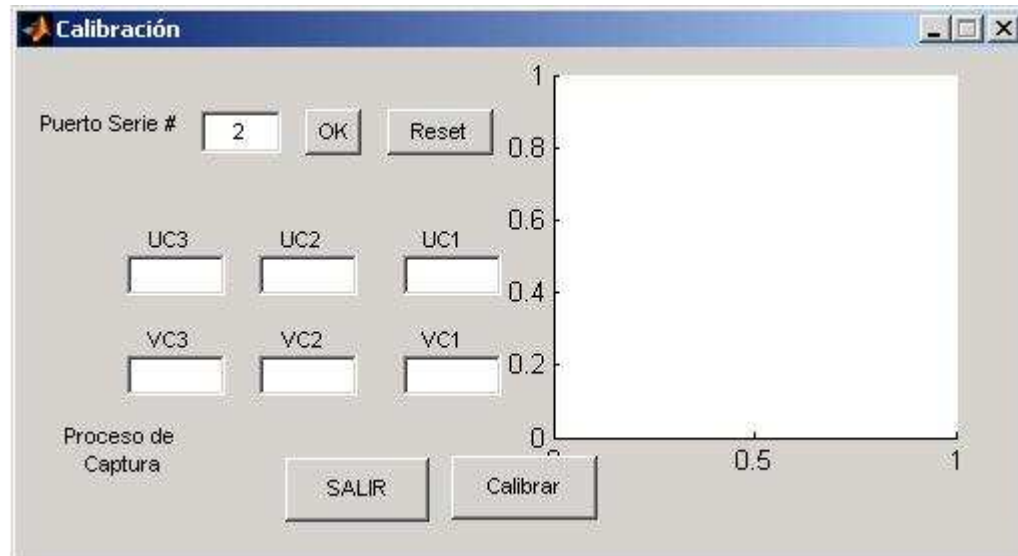


Fig. 113, Software de calibración.

Es el programa que obtiene la curva de calibración en U y la curva de calibración en V. Puede ser utilizado cuando la cámara y el láser hayan sido reemplazados o hayan cambiado su posición inicial en forma considerable.

Una vez se determine el puerto donde está conectado el circuito de control del digitalizador, se haya ubicado la cuadrícula en el centro de la base de escaneo y se presione el botón "Calibrar":

- La articulación de sensado se acercará 2mm a la cuadrícula por medio de un motor paso a paso que se encuentra dentro del eslabón del digitalizador.
- Se activa el láser, y la cámara tomará y guardará la foto de la cuadrícula.

Esto se repetirá hasta que la articulación de sensado haya tomado las muestras necesarias para obtener las curvas de calibración. Por último:

- Comienza el procesamiento de imágenes y luego el análisis matemático.

Terminado el proceso, se mostrarán los coeficientes que se guardaran en un archivo en formato MAT.

UC3, UC2 y UC1 son los coeficientes de los términos de orden dos, uno y cero respectivamente, de la curva de conversión en U. VC3, VC2 y VC1 son los coeficientes de la curva de conversión en V, en el mismo orden.

14.1.3 Software de análisis.



Fig. 114, Software de análisis.

Este software está diseñado para graficar el objeto digitalizado o determinadas franjas longitudinales y transversales del mismo. Pero MATLAB no solo se limita a dibujar la grafica del corte, sino que permite obtener datos de ésta, para realizar medidas que son difíciles de obtener por métodos convencionales.

El programa carga el archivo de nube de puntos en formato MAT, luego el usuario debe digitar el número de la franja, seleccionar el tipo de corte que quiere graficar y presionar el botón "Graficar". Si el corte seleccionado es longitudinal también puede dibujar la franja opuesta.

14.2 DIGITALIZADOR Y SUS PARTES FÍSICAS



Fig. 115, Digitalizador.



Fig. 116, Digitalizador y puerta deslizable.

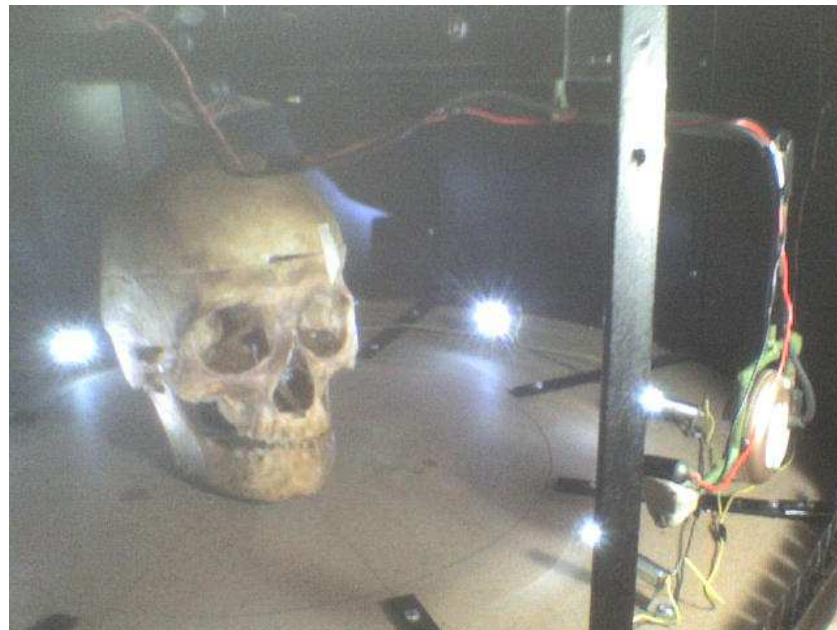


Fig. 117, Parte interior del digitalizador.

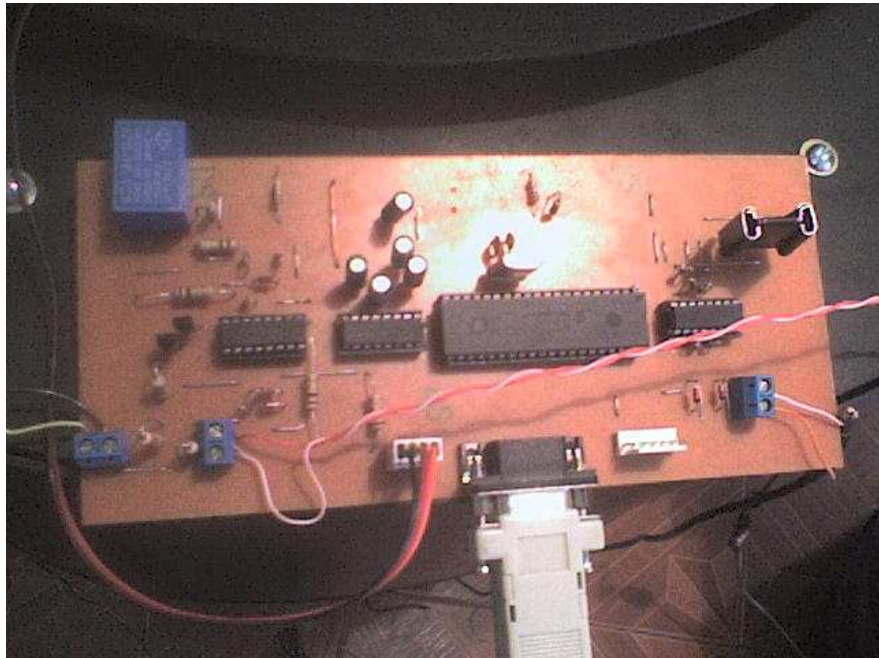


Fig. 118, Placa de control.



Fig. 119, Eslabón de de desplazamiento radial y angular.

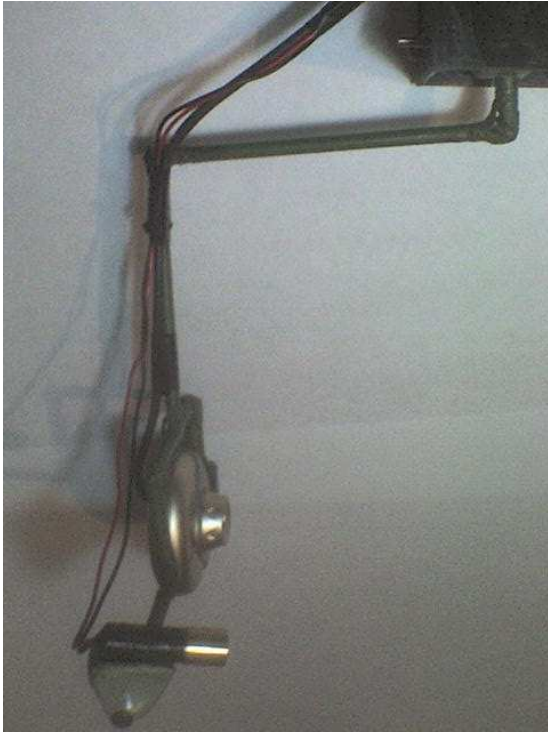


Fig. 120, Cámara y Láser.

15. CONCLUSIONES

Se desarrolló un sistema de adquisición de coordenadas espaciales que corresponden a puntos de la superficie de un objeto, basado en el principio de triangulación y utilizando luz estructurada y un sensor de imágenes que captura la deformación del patrón de luz cuando éste realiza un barrido alrededor del objeto. El sistema tiene características que le permiten una adecuada representación del objeto como una resolución de profundidad de 1.06mm, una resolución de altura de 0.34mm, un error del instrumento de 0.92mm, un rango de medida amplio y una densidad de la nube de puntos de 128.000 puntos o mayor, dándole un alto grado de exactitud y confianza. Sin embargo, existen limitantes en objetos con concavidades o agujeros, debido al principio en el cual se basa.

La calidad depende de factores como la iluminación, la reflectividad del objeto y su rugosidad. Hay materiales que reflejan gran cantidad de luz, otros que absorben gran parte del haz, incluso otros que esparcen o distorsionan el haz por su superficie debido a la rugosidad de ésta. Por lo tanto, en la calidad de la reconstrucción influye el grado de opacidad del objeto, suavidad de la superficie y color de ésta. Así, los mejores resultados se obtuvieron para objetos totalmente opacos, de colores claros y superficies suaves.

Los resultados obtenidos con la sustracción entre la imagen del objeto con patrón de luz y la imagen sin éste, realizado para cada incremento en el desplazamiento angular, fue exitoso porque redujo el ruido de fondo producido por fuentes de iluminación externas. Sin embargo no reduce por completo los efectos de la iluminación externa cuando incide sobre la proyección láser ya que hace que ésta no sea captada por la cámara. Para solucionar el problema, se cubrió el objeto con un material oscuro que lo aisló lumínicamente del ambiente.

Se trabajó con imágenes en escala de grises porque hay una buena diferenciación de intensidad entre el fondo de la imagen y la proyección del plano láser sobre el objeto, facilitándose la umbralización. Al trabajar con la trama de rojo de las imágenes RGB, la umbralización se dificulta porque las intensidades de rojo más altas no corresponden a la proyección del plano láser sino a reflexiones secundarias de éste sobre el objeto.

Con el pre-procesamiento digital de imágenes, para lo cual se utilizó una transformación de intensidad, se amplió el contraste entre el fondo de las imágenes y la proyección del plano láser sobre el objeto, mejorando el resultado de la umbralización y de la binarización.

Las operaciones y procesos morfológicos aplicados permiten obtener una curva de un píxel de ancho, ubicada en el centro de la proyección del plano láser sobre el objeto, siendo ésta una muy buena aproximación debido a que el generador láser tiene una distribución de intensidad gaussiana.

Con el principio de triangulación se obtuvieron de forma experimental dos curvas de conversión de píxeles a milímetros, que junto con la posición angular, permiten obtener las coordenadas cartesianas espaciales correspondientes a puntos de la superficie del objeto.

La utilización de un plano láser como patrón de iluminación permite escanear relativamente rápido un objeto, obteniendo una adecuada densidad de puntos (10 minutos para una densidad máxima de 128.000 puntos). Este tiempo de escaneo es mucho menor que el que necesitaría un sistema que utilice como patrón un punto láser, debido a que éste escanearía punto por punto, mientras que con el plano láser se obtiene información de toda una sección del objeto.

La información cromática permite identificar con facilidad partes del objeto digitalizado y es de gran importancia en aplicaciones donde la presentación del modelo es el principal objetivo como en museos virtuales.

El sistema de adquisición de información 3d es capaz de adquirir la geometría de objetos, reconstruirlos y visualizarlos como un modelo CAD, sirviendo como punto de partida para nuevas investigaciones en el campo de la visión por computador, siendo éste un tema nuevo y con un gran espectro de aplicaciones en la industria.

Los digitalizadores 3d se emplean en distintos campos de la industria para el control de calidad de piezas. Estos instrumentos se acoplan a bandas transportadoras que hacen pasar a cada objeto por el proceso de escaneo. La cámara y el láser se ubican en la parte superior de la banda y escanean la superficie que queda expuesta a estos dos elementos. El sistema realizado va encaminado a la digitalización de objetos abarcando 360° que permite obtener más cantidad de información debido al mayor campo de acción, haciéndolo muy útil en la animación tridimensional y la reproducción de obras de arte.

16. RECOMENDACIONES

El trabajo de investigación desarrollado sirve como base para posteriores trabajos en el campo de visión por computador y digitalización tridimensional. En él se presenta el proceso realizado desde la selección de los elementos de trabajo hasta el software de control realizado y el de visualización empleado. No obstante cada proyecto tiene sus respectivas especificaciones y delimitaciones, las cuales no deben ser pasadas por alto. A continuación se presentan algunas recomendaciones que pueden ser de gran utilidad en futuros trabajos:

El plano láser realiza un adecuado barrido alrededor del objeto, obteniéndose un tiempo de escaneo adecuado. Sin embargo hay patrones de iluminación como franjas láser, que permitirían reducir considerablemente el tiempo de escaneo, siendo esto muy importante en procesos donde el objeto a digitalizar no pueda permanecer mucho tiempo en una misma posición debido a la fatiga como es el caso de la digitalización de rostros.

La resolución del sistema de medidas depende en gran medida de la resolución de la cámara utilizada, debido a que hay mayor cantidad de píxeles que detectan cambios en la posición de la proyección láser. Por este motivo, se recomienda utilizar cámaras de alta resolución para obtener mejores resultados en la reconstrucción de objetos (mayor calidad en los detalles, textura, formas complejas etc.).

No se debe permitir que la iluminación ambiente incida sobre el objeto porque afecta el pre-procesamiento y posteriormente el procesamiento digital de imágenes, obteniéndose reconstrucciones erróneas. Para evitarlo se debe realizar un aislamiento lumínico entre el interior del digitalizador y el ambiente.

BIBLIOGRAFIA

GONZALEZ, Rafael. *Digital Image Processing Using Matlab*. Prentice Hall. Capítulos 2,3,9,10.

LÓPEZ GUERRERO, F. Eugenio. *Diseño de un instrumento para captura de geometrías 3D con aplicación en la manufactura de productos*. Pág. 2

Operaciones morfológicas básicas. Cap5.pdf

PATÍÑO VANEGAS, Alberto. *Sistema óptico para la extracción de parámetros metrológicos del pie*. Universidad Industrial de Santander. Págs. 16, 17, 18,19.

PÉREZ CORTÉS, Juan Carlos. *Los sistemas de inspección visual automática en los procesos industriales*. Universidad Politécnica de Valencia. Págs. 9, 10, 12, 13.

QUINTANILLA, Alberto. *Aplicación de un escáner 3d a fines didácticos*. Universidad de Valladolid. Págs. 2,3.

SOANCATL AGUILAR, Venustiano. *Recuperación de Información tridimensional usando luz estructurada*. Pág. 7.

www.digitalfotored.com/imagendigital/sensorimagen.htm

www.elai.upm.es/spain/Investiga/GCII/areas/luz/luz%20estructurada.htm

iabot.iespana.es/ciencia/software/ia/reconocimiento_de_imagenes.htm

www.interempresas.net/MetalMecanica/Articulos/Articulo.asp?A=8697

www.plastunivers.com/Arees/enginversa/inspeccion.htm

www.stockeryale.com/i/lasers/structured_light.htm

www.tsc.uc3m.es/Imagine/Curso_ProcesadoMorfologico/index.html

ANEXOS

ANEXO A. PLACA DEL CIRCUITO DE CONTROL

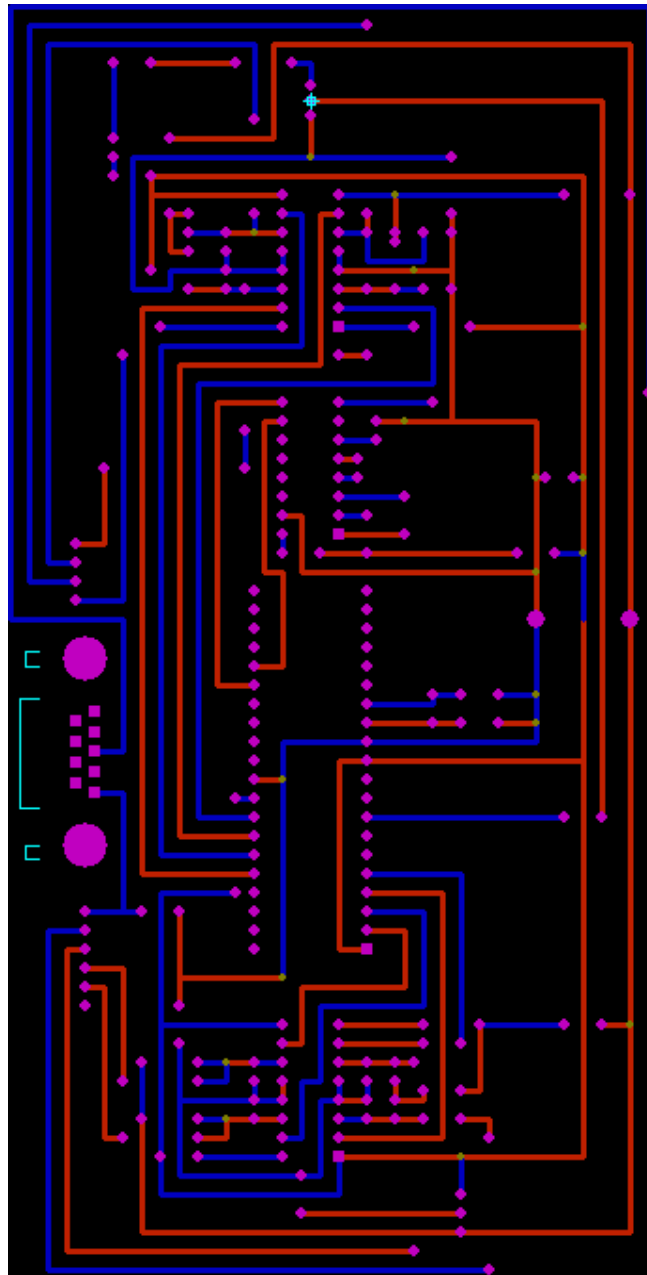
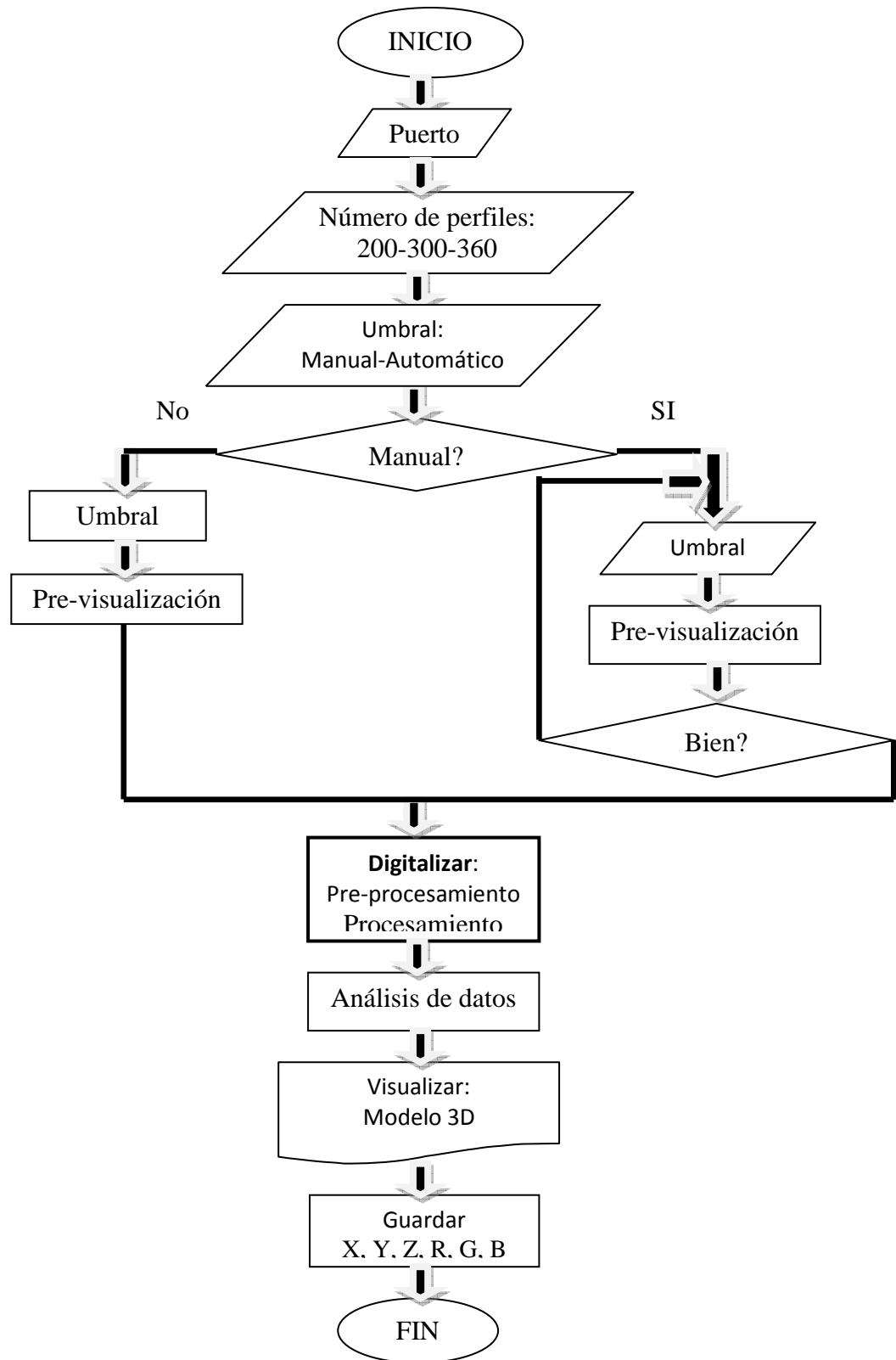
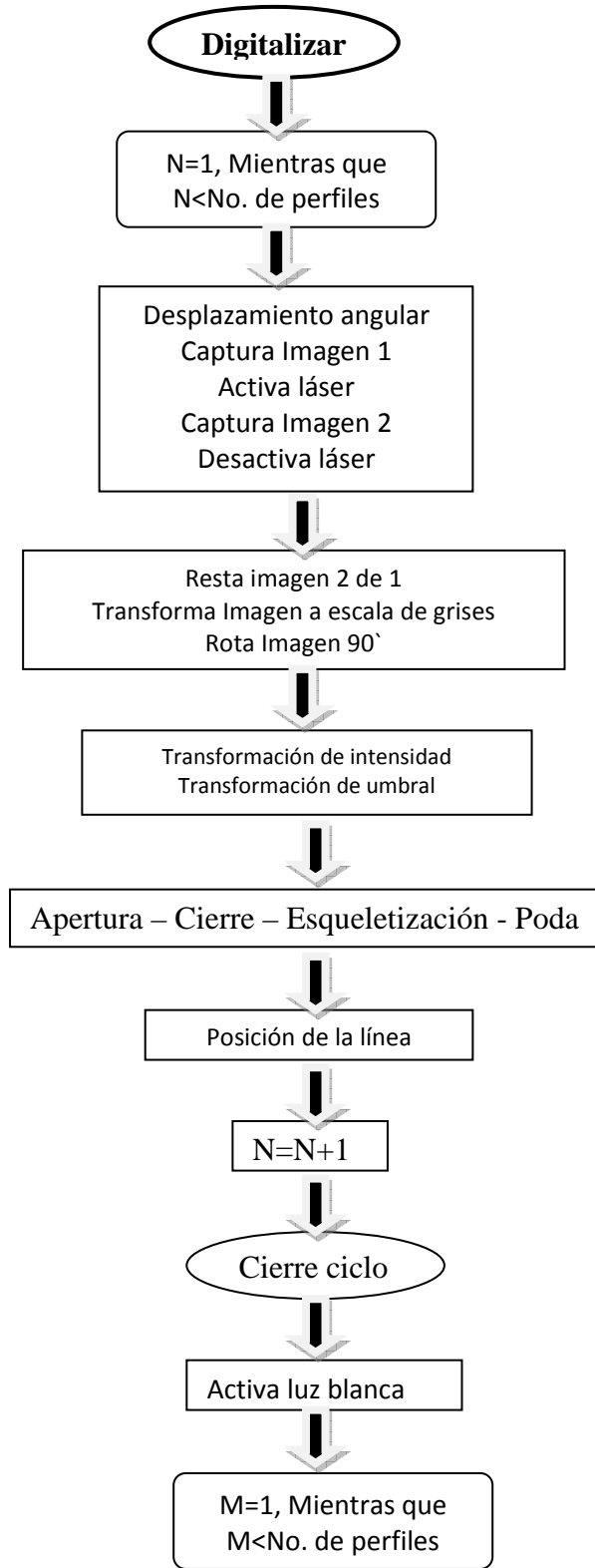


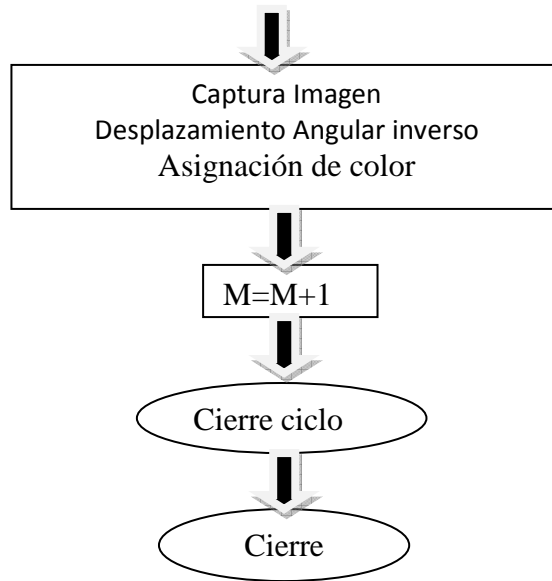
Fig. 121, Placa del circuito de control.

Anexo B. Diagrama de flujo Principal



Anexo C. Diagrama de flujo de la función Digitalizar





Anexo D. Código Fuente del programa de Digitalización

```
function varargout = onceget(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @onceget_OpeningFcn, ...
                  'gui_OutputFcn', @onceget_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

function onceget_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

set(handles.pushbutton1,'Enable','off') %Condiciones iniciales de los botones del
programa
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton5,'Enable','off')
set(handles.pushbutton7,'Enable','off')
set(handles.pushbutton10,'Enable','off')
set(handles.edit5,'Enable','off')

global aa sea seb sec w stnm sts deg auto u2 u1 u0 v2 v1 v0r

imaqreset
try
aa=videoinput('winvideo',1,'rgb24_640x480');
set(aa,'FramesPerTrigger',1,'TriggerRepeat',Inf);
triggerconfig(aa,'manual');
start(aa);
catch
```

```

msgbox('NO HAY CÁMARA CONECTADA')
close gcbf
end

u2=-0.002; u1=1.0496; u0=2.3784;
v2=-1.6364e-7; v1=-0.0011; v0=0.3476;
auto=0;      w=1; %contador de pasos
set(handles.text6,'String','1')
stnm=200; sts=9; deg=1.8;

sea=strel('square',3);
seb=strel('square',20);
sec=strel('line',2,0);
guidata(hObject, handles);
function varargout = onceget_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
function pushbutton1_Callback(hObject, eventdata, handles) %PREVISUALIZACIÓN
set(handles.pushbutton1,'Enable','off')
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton6,'Enable','off')
set(handles.edit1,'Enable','off')
set(handles.pushbutton10,'Enable','off');
set(handles.radiobutton6,'Enable','off')
set(handles.radiobutton7,'Enable','off')

global aa sea seb sec s2 auto ini
ini=640;
s=serial(s2);
s.BaudRate=2400;
fopen(s); %Abre puerto

trigger(aa);
f=getdata(aa); %Foto Uno
fprintf(s,'e');
pause(0.4)
trigger(aa);
ff=getdata(aa); %Foto Dos
fprintf(s,'f');

gr=rot90(ff(:, :, 1));
gg=rot90(ff(:, :, 2));
gb=rot90(ff(:, :, 3));

```

```

gt(:,:,1)=gr;
gt(:,:,2)=gg;
gt(:,:,3)=gb; %Giro Imagen RGB

axes(handles.axes1);
background=gt;
axis off
imshow(background); % Previsualización Imagen RGB

h90=imabsdiff(f,ff);
hn90=rgb2gray(h90);
nn90=rot90(hn90); %Giro Imagen en escala de gris

ffi=480;      hh=0; %UMBRAL
gg=imadjust(nn90,[],[],2); %Transformación de Intensidad
if auto == 1 %Automático
    in=ini/5;
    for we=1:5
        ngg(1:in,1:ffi)=gg(in*(we-1)+1:in*we,1:ffi);
        [mm mn]=imhist(ngg,256); %Histograma y análisis
        mnn=mm(30:256);
        [re po]=max(mnn);
        res=po+29;
        umbral(we)=res-res*0.2;
        st(1:in,1:ffi)=1./(1+(umbral(we)./(double(ngg)+eps)).^20);
        hh(in*(we-1)+1:in*we,1:ffi)=im2bw(st);
    end
    umbral=mean(umbral);
    set(handles.edit1,'String',umbral)
elseif auto == 0 %Manual
    umbral=str2num(get(handles.edit1,'String'));
    st=1./(1+(umbral/(double(gg)+eps)).^20);
    hh=im2bw(st);
end

axes(handles.axes2);
background=hh;
axis off
imshow(background) %Previsualización Imagen con Umbral

hoc=imclose(hh,seb);
hsk=bwmorph(hoc,'skel',Inf);

```



```
hsk=bwmorph(hsk,'spur',4);
hfin=bwmorph(hsk,'clean'); %Procesamiento de Imagen binaria
```

```
axes(handles.axes3);
background=hfin;
axis off
imshow(background) %Previsualización Imagen Esqueletizada
```

```
fclose(s)
delete(s) %Cierre del puerto
```

```
set(handles.pushbutton1,'Enable','on')
set(handles.pushbutton2,'Enable','on')
set(handles.pushbutton6,'Enable','on')
```

```
if auto == 0
set(handles.edit1,'Enable','on')
end
```

```
set(handles.pushbutton10,'Enable','on')
set(handles.radiobutton6,'Enable','on')
set(handles.radiobutton7,'Enable','on')
guidata(hObject,handles)
```

```
function pushbutton2_Callback(hObject, eventdata, handles) %DIGITALIZACIÓN
```

```
set(handles.pushbutton2,'Userdata',1)
set(handles.pushbutton1,'Enable','off')
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton3,'Enable','on')
set(handles.pushbutton4,'Enable','on')
set(handles.pushbutton7,'Enable','off')
set(handles.pushbutton10,'Enable','off')
set(handles.radiobutton1,'Enable','off')
set(handles.radiobutton2,'Enable','off')
set(handles.radiobutton3,'Enable','off')
set(handles.edit1,'Enable','off')
set(handles.radiobutton6,'Enable','off')
set(handles.radiobutton7,'Enable','off')
```

```
global aa sea seb sec w exvar s poscol posfil stnm sts x y z s2 auto rl gl bl rll gll bll sig ini
```

```
Xwv=stnm+1;
```

```
x=0; y=0; z=0; rll=0; gll=0; bll=0; sig=0;
poscol=0; posfil=0;
```

```
s=serial(s2); s.BaudRate=2400;
fopen(s); %Abre puerto
exvar=0;
```

```
while w < (stnm+1) %Número de desplazamientos angulares
```

```
    if w==1
        pause(1.4)
    end
```

```
    for k=1:1:sts %Número de pasos
```

```
        if (get(handles.pushbutton2,'Userdata') == 0 && AAA == 0), Xwv=w;
            w=stnm; AAA=1; break, end
```

```
        fprintf(s,'b') %Paso del motor
        pause(0.06)
    end
```

```
        if (get(handles.pushbutton2,'Userdata') == 0 && AAA == 0), Xwv=w; w=stnm;
            AAA=1; break, end
```

```
        if (get(handles.pushbutton2,'Userdata') == 0 && AAA == 1), break, end
```

```
    fprintf(s,'e'); %Activación del láser
    pause(0.3);
    trigger(aa)
```

```
        if (get(handles.pushbutton2,'Userdata') == 0 && AAA == 0), Xwv=w; w=stnm;
            AAA=1; break, end
```

```
        if (get(handles.pushbutton2,'Userdata') == 0 && AAA == 1), break, end
```

```
    ff=getdata(aa); %Captura Imagen
    fprintf(s,'f'); %Desactiva láser
    ini=640;
    hn90=rgb2gray(ff);
    nn90=rot90(hn90(:,1:ini));
    ffi=480; hh=0;
```

```
    gg=imadjust(nn90,[],[],2); %Pre-procesamiento
```

```
    if auto == 1 %Umbral Automático
        in=ini/5;
```

```

for we=1:5
ngg(1:in,1:ffi)=gg(in*(we-1)+1:in*we,1:ffi);
[mm mn]=imhist(ngg,256);
mnn=mm(30:256);
[re po]=max(mnn);
res=po+29;
umbral(we)=res-res*0.2;
st(1:in,1:ffi)=1./(1+(umbral(we)./(double(ngg)+eps)).^20);
hh(in*(we-1)+1:in*we,1:ffi)=im2bw(st);
end
umbral=mean(umbral);
set(handles.edit1,'String',umbral)
elseif auto == 0 %Umbral Manual
umbral=str2num(get(handles.edit1,'String'));
st=1./(1+(umbral)./(double(gg)+eps)).^20);
hh=im2bw(st);
end

hoc=imclose(hh,seb);
hsk=bwmorph(hoc,'skel',Inf);
hsks=bwmorph(hsk,'spur',4);
hfin=bwmorph(hsks,'clean'); %Procesamiento
[ma(w,1:ini),pos(w,1:ini)]=max(rot90(hfin)); %Posición de la línea

set(handles.text6,'string',s2)
drawnow
set(handles.text3,'string',w)
drawnow

axes(handles.axes1);
background=nn90;
axis off
imshow(background) %Visualización Imagen en escala de gris

axes(handles.axes2);
background=hh;
axis off
imshow(background) %Visualización Imagen con Umbral

axes(handles.axes3);
background=hfin;
axis off

```

```

    imshow(background) %Visualización Imagen Esqueletizada
    w=w+1;
end

set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
foto=1;

while((get(handles.pushbutton2,'UserData') == 1) && (foto < w-1))
for foto=1:1:(w-1)
    c=1;
    for fila=1:1:ini
        if pos(foto,fila)==1
            else if pos(foto,fila)~=1
                poscol(foto,c)=pos(foto,fila); % Posición línea (columna)
                posfil(foto,c)=fila; %Posición línea (fila)
                c=c+1;
            end
        end
    end
    set(handles.text3,'string',foto)
    drawnow
end
end

w=Xwv; %w RECUPERA SU VALOR ORIGINAL
if w == stnm+1 %SOLO PARA ESCANEADO EXITOSO, DEVUELVE CAMARA
    set(handles.pushbutton6,'Enable','off')
    fprintf(s,'g') %Activación de luz blanca
    pause(1)
    for reg=1:1:(w-1)
        trigger(aa) %Captura Imagen
        ccc=getdata(aa);
        gr=rot90(ccc(:,:,1),2);
        gg=rot90(ccc(:,:,2),2);
        gb=rot90(ccc(:,:,3),2);
        for co=1:1:length(poscol(w-reg,:)),
            if poscol((w-reg),co)==0 || posfil((w-reg),co)==0
                rl((w-reg),co)=1;
                gl((w-reg),co)=1;
                bl((w-reg),co)=1;
            end
        end
    end
end

```

```

        rl((w-reg),co)=gr(poscol((w-reg),co),posfil((w-reg),co));
        gl((w-reg),co)=gg(poscol((w-reg),co),posfil((w-reg),co));
        bl((w-reg),co)=gb(poscol((w-reg),co),posfil((w-reg),co)); %COLOR de los puntos
    end
end

for k=1:1:(0+sts) %Número de pasos
    fprintf(s,'a') %Pasos del motor en sentido contrario
    pause(0.06)
end
set(handles.text3,'string',reg)
drawnow
end

fprintf(s,'h') %Salidas del microcontrolador a cero

w=1;
fclose(s)
delete(s) %Cierre del Puerto
exvar=1;
set(handles.pushbutton5,'Enable','on')
set(handles.pushbutton6,'Enable','on')
set(handles.pushbutton1,'Enable','on')
set(handles.pushbutton2,'Enable','on')
set(handles.pushbutton10,'Enable','on');
set(handles.radiobutton6,'Enable','on')
set(handles.radiobutton7,'Enable','on')
end

msgbox('Listo!','WindowStyle','modal')
guidata(hObject, handles);

function pushbutton3_Callback(hObject, eventdata, handles) %PAUSA
global s exvar w
set(handles.pushbutton2,'UserData',0);
set(handles.pushbutton3,'eNABLE','Off');
set(handles.pushbutton4,'Enable','off');
pause(0.06)
fprintf(s,'h')
fclose(s)
delete(s) %Cierre del Puerto
exvar=1;

```

```
set(handles.pushbutton2,'ENABLE','ON');
guidata(hObject, handles);
```

```
function pushbutton4_Callback(hObject, eventdata, handles) %DETENER
```

```
global s w exvar sts
```

```
set(handles.pushbutton2,'UserData',0);
set(handles.pushbutton3,'Enable','off');
set(handles.pushbutton4,'Enable','off');
set(handles.pushbutton6,'Enable','off');
pause(0.06)
```

```
for reg=1:1:(w-1)
    if reg == 1
        for k=1:1:(0+sts) %Número de pasos
            fprintf(s,'a') %Pasos en sentido contrario
            pause(0.06)
        end
    else
        for k=1:1:sts %Número de pasos
            fprintf(s,'a') %Pasos en sentido contrario
            pause(0.06)
        end
    end
    set(handles.text3,'string',reg)
    drawnow
end
```

```
fprintf(s,'h') %Salidas del microcontrolador en cero
w=1;
```

```
fclose(s)
```

```
delete(s) %Cierre del puerto
```

```
set(handles.pushbutton1,'Enable','on');
set(handles.pushbutton2,'Enable','on');
set(handles.pushbutton6,'Enable','on');
set(handles.pushbutton10,'Enable','on');
set(handles.radiobutton1,'Enable','on')
set(handles.radiobutton2,'Enable','on')
set(handles.radiobutton3,'Enable','on')
set(handles.radiobutton6,'Enable','on')
set(handles.radiobutton7,'Enable','on')
exvar=1;
guidata(hObject, handles);
```

```

function pushbutton5_Callback(hObject, eventdata, handles) %ANALISIS
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton5,'Enable','off')
set(handles.pushbutton7,'Enable','off')

global poscol posfil x y z deg stnm rl gl bl rll gll bll poscn posfn u2 u1 u0 v2 v1 v0 sig

figure('WindowStyle','modal')

poscn=0;    posfn=0;
poscol=poscol-256; %REFERENCIA
[zf zc]=size(posfil);
for fo=1:zf
c=1;
    for fi=1:zc
        if poscol(fo,fi)<-50
        else if poscol(fo,fi)>=-50
            if poscol(fo,fi)>0
                sig(fo,c)=1;
            else sig(fo,c)=0;
            end
            poscn(fo,c)=poscol(fo,fi);
            posfn(fo,c)=posfil(fo,fi);
            rll(fo,c)=rl(fo,fi);
            gll(fo,c)=gl(fo,fi);
            bll(fo,c)=bl(fo,fi);          %Procesamiento de datos
            c=c+1;
        end
    end
end
end

[zf zc]=size(poscn);

for foto=1:zf,
    for fila=1:zc,
        radio(foto,fila)=(u2*(poscn(foto,fila))^2+u1*abs(poscn(foto,fila))+u0)/cos((15*pi)/180);
    %RADIO
        x(foto,fila)=radio(foto,fila)*cos(((deg*foto)*pi/180)+sig(foto,fila)*pi); %COOR X
        y(foto,fila)=radio(foto,fila)*sin(((deg*foto)*pi/180+sig(foto,fila)*pi)); %COOR Y
    end
end

```

```

(foto,filas)=(v2*(radio(foto,filas))^2+v1*radio(foto,filas)+v0)*(posfn(foto,filas)+u0*radio(foto,filas));
%COOR Z
    end
end
z=max(z:)-z;
set(handles.pushbutton7,'Enable','on')
set(handles.edit5,'Enable','on')

L(:,1)=x(:); L(:,2)=y(:); L(:,3)=z(:); %Grafica a color
col(:,:,1)=uint8(rl);
col(:,:,2)=uint8(gll);
col(:,:,3)=uint8(bl);
[in,map]=rgb2ind(col,2000);
L(:,4)=in(:);
m=sortrows(L,4);
dLix = [find(diff(m(:,4))>0)];
hold on;
s=1;

for k = 1:length(dLix)
plot3(m(s:dLix(k),1),m(s:dLix(k),2),m(s:dLix(k),3),'o','MarkerSize',4,'Color',map(m(s,4)+1,:),'
MarkerEdgeColor',map(m(s,4)+1,:),'MarkerFaceColor',map(m(s,4)+1,:))
    s = dLix(k)+1;
set(gca,'DataAspectRatio',[1,1,1],'Color',[1,1,1],'XColor',[1,0,0],'YColor',[0,1,0],'ZColor',[0,0,
1]);
    view(-180,47)
    xlim([min(x(:)) max(x(:))])
    ylim([min(y(:)) max(y(:))])
    zlim([min(z(:)) max(z(:))])
    box on
    grid on
end
hold off;

close Figure 1

set(handles.radiobutton1,'Enable','on')
set(handles.radiobutton2,'Enable','on')
set(handles.radiobutton3,'Enable','on')
set(handles.edit1,'Enable','on')
guidata(hObject, handles);

```



```

function pushbutton6_Callback(hObject, eventdata, handles) %SALIR
pause(0.06)
set(handles.pushbutton1,'Enable','off')
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton3,'Enable','off')
set(handles.pushbutton4,'Enable','off')
set(handles.pushbutton5,'Enable','off')
set(handles.pushbutton6,'Enable','off')

global aa s exvar w sts

if exvar == 0
    for reg=1:1:(w-1)
        for k=1:1:sts
            fprintf(s,'a') %Pasos en sentido contrario
            pause(0.06)
        end
    end

    set(handles.pushbutton2,'UserData',0);
    fprintf(s,'h') %Salida del microcontrolador en cero
    fclose(s)
    delete(s)
end

stop(aa)
delete(aa) %Cierre de la cámara
close(gcf)

function edit1_Callback(hObject, eventdata, handles) %INGRESO UMBRAL
umb=get(hObject,'String');
umbn=str2num(umb);

if isempty(umbn)
    set(hObject,'String','120');
end
guidata(hObject,handles);

function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

end

```
function pushbutton7_Callback(hObject, eventdata, handles) %GUARDAR
global x y z rll gll bll poscn posfn
d=[x(:) y(:) z(:)];
rll=uint8(rll); gll=uint8(gll); bll=uint8(bll);
wy=get(handles.edit5,'String');
s=char(wy);
save(s,'x','y','z','rll','gll','bll','poscn','posfn');
txt='.txt';
rain=char(strcat(wy,txt));
fid=fopen(rain,'wt');
fprintf(fid, '%12.8f %12.8f %12.8f\n',d)
fclose(fid)
set(handles.edit5,'Enable','off')
set(handles.pushbutton7,'Enable','off')
guidata(hObject,handles)
```

```
function radiobutton1_Callback(hObject, eventdata, handles) %DESPLAZAMIENTO
ANGULAR 1.8
h=get(handles.radiobutton1,'Value');
global stnm sts deg
if h == 1
set(handles.text6,'String','1')
stnm=200; sts=9; deg=1.8;
end
guidata(hObject,handles)
```

```
function radiobutton2_Callback(hObject, eventdata, handles) %DESPLAZAMIENTO
ANGULAR 1.2
h=get(handles.radiobutton2,'Value');
global stnm sts deg
if h == 1
set(handles.text6,'String','2')
stnm=300; sts=6; deg=1.2;
end
guidata(hObject,handles)
```

```
function radiobutton3_Callback(hObject, eventdata, handles) %DESPLAZAMIENTO
ANGULAR 1
h=get(handles.radiobutton3,'Value');
global stnm sts deg
```

```

if h == 1
set(handles.text6,'String','3')
stnm=360; sts=5; deg=1;
end
guidata(hObject,handles)

function edit4_Callback(hObject, eventdata, handles)

function edit4_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushbutton9_Callback(hObject, eventdata, handles) %SELECCIÓN PUERTO
global s2
h=get(handles.edit4,'String');
s1='COM';
s2=strcat(s1,h);

set(handles.edit4,'Enable','off')
set(handles.pushbutton1,'Enable','on')
set(handles.pushbutton2,'Enable','on')
set(handles.pushbutton9,'Enable','off')%
set(handles.pushbutton10,'Enable','on')
guidata(hObject,handles)

function pushbutton10_Callback(hObject, eventdata, handles) %RESET
set(handles.edit4,'Enable','on')
set(handles.pushbutton1,'Enable','off')
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton9,'Enable','on')
set(handles.pushbutton10,'Enable','off')
guidata(hObject,handles)

function edit5_Callback(hObject, eventdata, handles)

function edit5_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```
function uipanel2_SelectionChangeFcn(hObject, eventdata, handles) %SELECCION
UMBRAL
global auto
if (hObject==handles.radiobutton6)%AUTO
    auto=1;
    set(handles.edit1,'Enable','off')
elseif (hObject==handles.radiobutton7)%MANUAL
    auto=0;
    set(handles.edit1,'Enable','on')
end
guidata(hObject,handles)
```