



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

CENTRO DE INFORMÁTICA INDUSTRIAL

ADQUISICIÓN DE MODELOS GEOMÉTRICOS A PARTIR DE OBJETOS DEL MUNDO REAL

Tesis presentada para optar por el título de Ingeniero en Ciencias Informáticas

Autor: Adrián Fonseca Mursulí

Tutor: Ing. Ernesto De La Cruz Guevara Ramírez

Co-tutor: Ing. Mileydi Moreno Mirabal

Ciudad de la Habana

Junio de 2011

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Adrián Fonseca Mursulí

Firma del Autor

DATOS DE CONTACTO

Tutor: Ing. Ernesto De La Cruz Guevara Ramírez.

Graduado como Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI) en el año 2008. Actualmente pertenece al Centro de Informática Industrial (CEDIN) de la Facultad 5 de la UCI.

Empresa: Universidad de Ciencias Informáticas UCI.

Dirección: Carretera a San Antonio Km. 2 1/2 Torrens, Boyeros La Habana.

e-mail: elquevara@uci.cu

Co-tutor: Ing. Mileydi Moreno Mirabal

Graduada como Ingeniera en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI) en el año 2008. Actualmente pertenece al Centro de Informática Industrial (CEDIN) de la Facultad 5 de la UCI.

Empresa: Universidad de Ciencias Informáticas UCI.

Dirección: Carretera a San Antonio Km. 2 1/2 Torrens, Boyeros La Habana.

e-mail: mmirabal@uci.cu

A mi familia... en especial a mi mamá, a mi abuela Cocha, a mi tío Carlos, a mi padre Alvio, a mi hermana Rosanna, a mi papá Alexis y a mi prima Mónica y a mi primo Frank David por darme todo el apoyo del mundo.

A mi novia Camila... por ser amiga, por ser la persona que ha estado a mi lado durante toda mi carrera.

A mis amigos... Adriel, Julio, Carlos Vallejo y Danae por ser mis hermanos y porque sé que siempre podré contar con ellos.

A mis amistades... Ivis, Alberto, Camelia, Yenier, Libertad, Arlen, Sonia, Yury, Odalis, Omar, Mayelin, Reynaldo, Lázaro y Abel Ernesto por valorar mi amistad y estar pendientes a mis problemas.

A mis compañeros...Liudmila, Yoana, Maybe, Yary, Nory, Yanet, Ray, Oniel, Siles, Lea, Palomino, Alejandro, Consuegra, Andy, Yuniel Ramos, Yaniel, Dayana, Yuniel Castro, Anay, Rubén, Yeilin y Guille por ayudarme y estar siempre dispuestos a ayudarme.

A mis tutores...Ernesto y Mileydi por su apoyo y dedicación.

A los que de una forma u otra me ayudaron con la realización de esta Tesis... Santiago y Liudmila Pupo.

A todos por los que pase por su lado y dejaron en mi camino un momento de felicidad y un grano de sabiduría.

Gracias, mil gracias a todos.

A mi mamá por ser la madre más especial del mundo y guiarme en todos los momentos de mi vida, a mi abuela Cocha por tanto cariño y enseñarme el camino de la vida, a mi abuelo Serafín porque se que estuviera muy orgulloso de mí, a mi tío Carlos por quererme tanto, a Rosanna, Arianna, Gabriel Enrique y Arhan por ser los hermanos más cariñosos, a mi padre Alvio y a mi papá Alexis por ser la guía a seguir, a mis otros abuelos, a mis primos, a mis tíos, a mis amigos, a mi familia que siempre ha estado y estará en mi corazón como el tesoro más grande que tengo.

La visión por computador es una rama de la inteligencia artificial que desarrolla tanto la teoría como las tecnologías necesarias para simular la percepción visual humana, su principal objetivo es construir sistemas artificiales capaces de interpretar escenas naturales o datos multidimensionales. En la actualidad la visión por computador se ha desarrollado rápidamente debido a la gran variedad de aplicaciones en la que estas informaciones son necesarias. El ser humano cuenta con un sistema visual muy complejo, no obstante, las computadoras que operan como sistemas visuales artificiales pueden medir de forma precisa los objetos que se encuentran en un entorno en lugar de interpretarlo. La extracción de estas medidas a partir de una o varias imágenes es el objetivo principal de la fotogrametría.

El objetivo de este trabajo es desarrollar un prototipo de software capaz de adquirir información geométrica de la superficie de un objeto real, mediante la fotogrametría como técnica de adquisición, obteniendo como resultado una nube de puntos tridimensionales a partir del análisis de las diferentes etapas del proceso de obtención de la geometría, que permite mejorar el trabajo de los diseñadores en cuanto tiempo de modelado, ahorro de esfuerzos y recursos, manteniendo su calidad visual, precisión y exactitud.

Palabras clave: Visión por computador, reconstrucción tridimensional, sistemas visuales artificiales, fotogrametría.

TABLA DE CONTENIDO

INTRODUCCIÓN 1

CAPÍTULO 1. Fundamentación Teórica..... 4

 Introducción 4

 1.1 Conceptos asociados al dominio del problema..... 4

 1.1.1 Sistemas de visión por computador 4

 1.1.2 Representación de imágenes digitales..... 5

 1.1.3 Proyección perspectiva..... 6

 1.1.4 Paralaje 6

 1.2 Etapas en un proceso de visión artificial 6

 1.3 Técnicas de adquisición de información 3D de una escena 7

 1.3.1 Técnicas ópticas..... 8

 Técnicas pasivas..... 9

 Técnicas activas 12

 1.3.2 Tabla comparativa entre técnicas de adquisición tridimensional..... 15

 1.4 Componentes de un sistema de visión por computador para la reconstrucción 3D..... 15

 1.4.1 Fuente de iluminación 16

 1.4.2 Cámara..... 16

 1.4.3 Óptica 16

 1.5 Calibración de un sistema de visión por computador 17

 1.6 Formatos de ficheros para la obtención de nube de puntos 19

 1.6.1 Formato de fichero OBJ 19

 1.6.2 Formato de fichero TXT 19

 1.6.4 Formato de fichero PLY 20

CAPÍTULO 2. Solución Propuesta..... 21

 Introducción 21

 2.1 Selección de la técnica de adquisición geométrica 21

 2.2 Etapas en el proceso de adquisición geométrica 22

 2.3 Adquisición de imágenes..... 23

 2.4 Modelo de cámaras. Geometría 3D 24

 2.4.1 Interpretación geométrica y matricial del modelo Pinhole 26

 2.4.2 Distorsiones geométricas 27

 Distorsión radial..... 28

2.4.3 Matriz intrínseca.....	29
2.5 Obtención de puntos de control entre dos cámaras que ven una misma escena	29
2.5.1 Geometría epipolar	29
2.5.2 Correspondencia de un punto en dos imágenes.....	31
2.5.3 Parámetros extrínsecos	32
2.6 Rectificación de imágenes	32
2.7 Cálculo de disparidad o profundidad	33
2.8 Obtención de nube de puntos.....	35
2.9 Recursos utilizados en el proceso de desarrollo	35
CAPÍTULO 3 Ingeniería del sistema	38
Introducción	38
3.1 Características del sistema	38
3.1.1 Patrones de arquitectura.....	38
3.1.2 Reglas del negocio	39
3.1.3 Modelo de domino o de negocio	39
3.1.4 Glosario de términos	40
3.1.5 Captura de requisitos funcionales.....	40
Requisitos funcionales.....	40
Requisitos no funcionales.....	41
3.1.6 Modelo de casos de uso del sistema.....	42
Actores del sistema	42
Diagrama de casos de uso del sistema.....	43
Descripción de casos de uso del sistema.....	43
3.2 Diseño del sistema.....	50
3.2.1 Patrones de diseño	50
3.2.2 Diagramas de clases del diseño del sistema.....	51
3.2.3 Descripción de las clases de diseño.....	52
Diagramas de interacción del diseño	56
Capítulo 4. Implementación y prueba	58
Introducción	58
4.1 Implementación.....	58
4.1.1 Diagrama de componente	58
4.2 Prueba	59
4.2.1 Fase de prueba	59

4.2.2 Recursos empleados en la realización de las pruebas	59
4.3 Pruebas desarrolladas	59
CONCLUSIONES	65
RECOMENDACIONES	66
BIBLIOGRAFÍA	67
ANEXOS.....	69
Anexo 1. Transformación de nube de puntos en modelos tridimensionales.	69
Anexo 2. Encuesta realizada a diseñadores del proyecto Escenarios 3D.....	69
Anexo 3. Opinión del jefe de la Línea de Imágenes y Videos basados en Rendering.	70

INDICE DE FIGURAS

Figura 1. Clasificación de técnicas de adquisición de información 3D de una escena.....	7
Figura 2. Clasificación de técnicas ópticas.....	9
Figura 3. Técnica de visión estéreo.. ..	10
Figura 4. Equipo fotográfico con dos cámaras para visión estéreo.....	12
Figura 5. Relación de triángulos semejantes en la técnica de triangulación.. ..	13
Figura 6. Modelado 3D con proyección de luz blanca estructurada.. ..	14
Figura 7. Parámetros de calibración de una cámara.....	18
Figura 8. Proceso fotográfico de la adquisición tridimensional.	21
Figura 9. Ejemplos de patrones de calibración.....	23
Figura 10. Geometría proyectiva.....	25
Figura 11. Modelo ideal de cámara, Pinhole.....	25
Figura 12. Proyección de un punto A en el espacio.	26
Figura 13. Corrección de la distorsión radial.....	28
Figura 14. Parábola del negativo.....	29
Figura 15. Geometría epipolar	30
Figura 16. Correspondencia de un punto en dos imágenes procedentes de un punto 3D común.....	31
Figura 17. Planos de imagen alineados.....	32
Figura 18. Normalización de un par estereoscópico.	33
Figura 19. Ejemplo de mapa de disparidad.	34
Figura 20. Modelo del dominio.	39
Figura 21. Diagrama de Caso de Uso del Sistema.	43
Figura 22. Diagrama de Clases del Diseño.....	51
Figura 23. Diagrama de Secuencia: CU Calibrar sistema	56
Figura 24. Diagrama de Secuencia: CU Generar imágenes sin distorsión radial.....	56
Figura 25. Diagrama de Secuencia: CU Generar imágenes rectificadas.....	57
Figura 26. Diagrama de Secuencia: CU Generar nube puntos 3D.....	57
Figura 27. Diagrama de Componente.....	58
Figura 28. Fotos para la calibración.....	60
Figura 29. Detención de esquinas.	61

Figura 30. Par estéreo del objeto.	61
Figura 31. Obtención de puntos de control.	61
Figura 32. Imágenes estéreos rectificadas.....	62
Figura 33. Mapa de disparidad para SAD = 21.	62
Figura 34. Nube de puntos generada para SAD = 21.....	63
Figura 35. Gráfico generado a partir de los resultados obtenidos de la pregunta dos de la encuesta.....	63
Figura 36. Gráfico generado a partir de los resultados obtenidos de la pregunta tres de la encuesta.	64
Figura 37. Software de representación de la nube de puntos.	69

INDICE DE TABLAS

Tabla 1. Tabla comparativa entre técnicas de adquisición tridimensional.	15
Tabla 2. Actor del sistema.	43
Tabla 3. CU Gestionar proyecto.....	45
Tabla 4. CU Calibrar Sistema.....	47
Tabla 5. CU Generar imágenes sin distorsión radial.....	48
Tabla 6. CU Generar imágenes rectificadas.....	49
Tabla 7. CU Generar nube de puntos 3D.....	49
Tabla 8. Descripción de la clase Proyecto	53
Tabla 9. Descripción de la clase RectificacionEstereo.	53
Tabla 10. Descripción de la clase CalibradorExtrinseco.	54
Tabla 11. Descripción de la clase CalibradorIntrinseco.....	54
Tabla 12. Descripción de la clase cvPuntosSinDistorsion.....	54
Tabla 13. Descripción de la clase DgvManege.....	55
Tabla 14. Descripción de la clase PbManeger.	55
Tabla 15. Caso de prueba: Adquisición de putos tridimensionales.....	62

INTRODUCCIÓN

La extracción de información a partir de imágenes es una tarea muy fácil de realizar para un ser humano, el cual desde muy temprana edad tiene la capacidad de ver y reconocer objetos en el ambiente que le rodea. El sentido de la vista es una de las formas dominantes para percibir el mundo, imitar su funcionamiento en una computadora, con base en la interpretación de las imágenes, es una tarea extremadamente difícil. Se podría explicar tal proceso a nivel biológico, ya que se sabe cómo la luz es transformada en la retina y estimula ciertas neuronas perceptivas. Sin embargo, aún no es claro cómo esta transformación causa que la mente reconozca los objetos. Contrario al nivel bajo que se maneja en el área biológica neuronal, el área psicológica intenta dar interpretaciones a un alto nivel de cómo funciona el sistema visual. Los modelos que se proponen en el área psicológica no proporcionan instrucciones de cómo construir un sistema artificial comparable con el sentido visual que tiene el ser humano. En síntesis, se puede decir que el ser humano vive con un sistema visual único, del cual aún no se conoce a detalle cómo funciona.

Los 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. Existen aplicaciones en diferentes ramas de la sociedad como puede ser en la industria para el control de calidad o la ingeniería inversa, en la medicina para la creación de prótesis, cirugía plástica o dental, en el arte para la reproducción de obras de arte y su implantación en museos virtuales, en el comercio electrónico para la presentación de los artículos en venta a través de navegadores de manera que los clientes puedan verlos en 3 dimensiones y girarlos; también se pueden utilizar estos sistemas en otros campos como pueden ser en la Robótica, para el reconocimiento de marcas para el guiado de robots y en el área de ingeniería biomédica, para la reconstrucción de estructuras anatómicas a partir de imágenes médicas.

La visión artificial constituye uno de los temas de investigación que posee en la actualidad un espectro amplio de posibles aplicaciones, y que en un futuro inmediato va a adquirir todavía una mayor relevancia. Muestra de ello son los múltiples esfuerzos que se dedican por parte de la industria cubana del software y su interés en el desarrollo de nuevas aplicaciones, constituyendo eslabón fundamental en la actual situación económica del país.

En el departamento de Realidad Virtual del Centro de Informática Industrial se encuentra la Línea de Imágenes y Videos basados en Rendering, encargada del diseño de todos los proyectos vinculados con la Realidad Virtual, en estos momentos se encuentra inmersa en un intenso trabajo debido a la gran demanda de modelos tridimensionales. Por esta razón la Línea se encuentra interesada en los avances en el área de la adquisición de modelos geométricos a partir de objetos del mundo real mediante técnicas de obtención geométrica, la cual tiene la necesidad de un

prototipo cubano con coste económico reducido que aporte beneficios en cuanto a la disminución de esfuerzos, tiempo y recursos en el diseño de modelos tridimensionales, y que además mantenga su calidad visual, precisión, exactitud y versatilidad, aplicables principalmente a objetos de alta complejidad de modelado.

Por lo antes expuesto queda identificado el **problema científico**:

- ¿Cómo adquirir la información geométrica de un objeto real, para disminuir esfuerzos, tiempo y recursos en el diseño de modelos tridimensionales?

Dada la **problemática** anteriormente expuesta y la demanda de forma creciente de aplicaciones de reconstrucción 3D, esta tesis enmarca como **objetivo general**:

- Desarrollar un prototipo de software capaz de adquirir información geométrica de la superficie de un objeto real.

Objeto de estudio: Reconstrucción 3D basada en la adquisición de la información tridimensional.

Campo de acción: Adquisición de los modelos geométricos a partir de objetos del mundo real.

Idea a defender: Con el desarrollo de un prototipo de software basado en la adquisición de la información geométrica de un objeto real, se disminuye el tiempo de modelado y se ahorran recursos humanos y materiales.

Para dar cumplimiento a los objetivos planteados se han tomado las siguientes tareas investigativas:

- Estudiar la bibliografía referente a los adelantos en sistemas de adquisición de la geometría, para estructurar el marco teórico.
- Caracterizar las diferentes técnicas de procesamiento y reconstrucción geométrica para seleccionar la más adecuada que de solución al problema.
- Implementar una solución basada en una técnica previamente analizada.
- Realizar pruebas para comprobar la factibilidad de la solución.

Además para todo el proceso de investigación y elaboración de este trabajo se tomó en cuenta la utilización de varios métodos científicos de investigación como:

- **Método Analítico-Sintético:** Con el objetivo de analizar las teorías, documentos e información, permitiendo la extracción de los elementos más importantes que se relacionan con el objeto de estudio.
- **Inductivo–deductivo:** Para luego de inducir una serie de conocimientos referentes a la adquisición geométrica, poder arribar a razonamientos que conlleven a la deducción de

conocimientos que puedan ser aplicables al problema a tratar.

- **Análisis histórico-lógico:** Permitirá analizar la trayectoria y evolución de las técnicas de adquisición geométrica de objetos reales, para así saber cuan desarrollado está el tema a nivel global y tomar la decisión de cuál o cuáles desarrollar.
- **Experimento:** Método empírico mediante el cual, se realizarán experimentos y pruebas de los principales algoritmos para la adquisición de geometrías, para examinar los resultados en busca de escoger los más adecuados.
- **Observación:** Mediante este método empírico se observarán los resultados obtenidos en la caracterización e identificación de los principales algoritmos utilizados, para poder decidir luego cuál o cuáles serán más adecuados.

Esta tesis está organizada en 4 capítulos, detallados a continuación:

Capítulo 1:- Fundamentación Teórica. Se indican las bases teóricas fundamentales de las técnicas y algoritmos para la adquisición de modelos geométricos existentes en la actualidad, se identifican además componentes de estos sistemas de adquisición tridimensional.

Capítulo 2:- Solución Propuesta. Se describe la solución técnica del problema, teniendo en cuenta las técnicas y algoritmos escogidos, mediante la propuesta de las etapas fundamentales en el proceso de adquisición tridimensional.

Capítulo 3:- Ingeniería del Sistema. Se realiza el levantamiento de requisitos funcionales y se modelan las características del análisis y diseño del sistema a través de los diagramas correspondientes.

Capítulo 4:- Implementación y Prueba. Se modela la implementación del sistema a través de los diagramas correspondientes y se muestran las pruebas realizadas al prototipo en aras de resolver los problemas planteados.

Finalmente se presentan las conclusiones y recomendaciones obtenidas gracias al resultado de la investigación.

CAPÍTULO 1. Fundamentación Teórica

Introducción

En este capítulo se abordarán los principales elementos teóricos que conforman tanto a los sistemas como a las técnicas de visión por computadora, que permiten obtener información tridimensional a partir de imágenes bidimensionales adquiridas de una escena. Se presta especial atención a las técnicas ópticas, y dentro de estas, a las pasivas, ya que son las más interesantes desde el punto de vista de la reconstrucción tridimensional y las más utilizadas en el ámbito social.

1.1 Conceptos asociados al dominio del problema

Para mejor entendimiento y desenvolvimiento de las áreas temáticas que se abordarán en el presente y posteriores capítulos, se mostrarán una serie de conceptos identificados durante la investigación realizada, así como la descripción de cada uno facilitando la comprensión de los temas abordados.

1.1.1 Sistemas de visión por computador

La visión por computador, también denominada visión artificial, visión de máquina, visión de robot, visión computacional, análisis de imágenes o interpretación de escenas, es el proceso de extracción de información del mundo real a partir de imágenes utilizando como herramienta una computadora. Desde un punto de vista técnico, un sistema de visión por computador es un sistema autónomo que realiza alguna de las tareas que el sistema de visión humano es capaz de realizar (1). Este sistema es capaz de extraer o deducir la estructura y las propiedades del mundo tridimensional a partir de una o más imágenes.

La capacidad de simular el efecto de la visión humana a través de un sistema de percepción electrónico no es una tarea sencilla, el principal obstáculo se encuentra en que el ojo humano percibe la luz reflejada por los objetos en un espacio de tres dimensiones (3D), mientras que las computadoras analizan estos objetos a partir de proyecciones en imágenes de dos dimensiones (2D). La reducción de una dimensión produce la pérdida de gran cantidad de información, lo que eleva notablemente la dificultad del proceso de visión. Las imágenes manejadas por las computadoras son bidimensionales debido a que el proceso de muestreo de la escena del mundo real se realiza, normalmente, a través de una rejilla rectangular, donde cada uno de sus elementos se conoce como píxel. Cada píxel puede almacenar valores de un tipo concreto, dependiendo del formato de representación de la imagen.

1.1.2 Representación de imágenes digitales

La representación de la información de una imagen se puede realizar de varias formas: a través de imágenes de intensidad, de imágenes térmicas o de imágenes de rango.

En las **imágenes de intensidad**, cada píxel representa el brillo o intensidad lumínica que ha detectado cada elemento fotosensible del sensor durante el período de exposición.

Las **imágenes térmicas**, generadas por sensores infrarrojos, indican la temperatura que emite cada cuerpo de la escena. En este caso, cada píxel de la imagen representa el valor de temperatura concreto que tiene un punto de la escena.

Las **imágenes de rango**, también denominadas imágenes de alcance, imágenes de profundidad, mapas de profundidad, mapas XYZ o perfiles de superficie, son la forma más común de representar las coordenadas tridimensionales de los objetos de una escena. Los dispositivos capaces de realizar este tipo de mediciones se denominan sensores de rango. En las imágenes de rango, cada píxel representa la distancia del punto de la escena a un sistema de referencia conocido. Si la referencia se toma con respecto a un sistema de coordenadas ortogonales situado en la escena, cada píxel suele representar una terna de valores que se corresponden con las coordenadas cartesianas absolutas de cada punto de la escena.

Independientemente del formato de representación, se tiene una matriz de puntos de dimensiones $M \times N$, es decir, una imagen digital, que debe ser tratada por un computador.

La segmentación está basada en dos principios fundamentales: discontinuidad o segmentación orientada a bordes y similitud, o segmentación orientada a regiones (3). Esta operación implica un procesamiento que supone una transformación de la imagen original, de forma que los valores de los píxeles originales son modificados mediante ciertas funciones de transformación u operadores. En ocasiones ocurre que la imagen no es lo suficientemente buena para extraer la información necesaria de forma adecuada, por lo que es necesario disponer de ciertas técnicas de mejora de la calidad de la imagen original.

Las técnicas de mejora de la calidad de una imagen se dividen en dos grandes grupos: suavizado, encaminado a la supresión del ruido introducido durante la captura de la imagen, y realzado, encaminado a eliminar falsos reflejos y sombras. Una vez detectados los bordes o localizadas las diferentes regiones de la imagen, es decir, sus elementos de interés, el siguiente paso consiste en extraer las propiedades o atributos que serán suministrados al componente de software del sistema de visión.

1.1.3 Proyección perspectiva

La proyección perspectiva, también denominada proyección central o transformación perspectiva, es la proyección de puntos tridimensionales del espacio (la escena) en puntos bidimensionales de un plano (la imagen) por medio de líneas de proyección que pasan a través de un punto, denominado centro de proyección o centro óptico (4).

1.1.4 Paralaje

Si un objeto cercano se observa alternativamente con el ojo derecho y el izquierdo, su posición parecerá cambiar. Este desplazamiento aparente, causado por un cambio en el punto de observación, es conocido como paralaje (5). Como se infiere de conceptos anteriormente vertidos, la existencia de paralaje es una característica normal de las fotografías con recubrimiento, siendo la base de la visión estereoscópica. La sensación de relieve en la observación de fotos aéreas, es debida a esos desplazamientos de imagen. Para la determinación de la distancia de los objetos observados en las fotos, se podrá medir esas diferencias de posición, o sea la paralaje.

Paralaje estereoscópica absoluta (o Paralaje X), es la diferencia de las distancias entre los puntos principales y las imágenes homólogas, medidas paralelas a la línea de vuelo (componente X). Paralaje relativa o diferencial es simplemente la diferencia de paralajes absolutas entre dos puntos.

El hecho de encontrarse el objeto a una distancia finita de las cámaras, arroja diferentes posiciones en las fotografías. De encontrarse a distancia infinita se tendrá la misma posición en ambas fotos, al ser paralelos los rayos homólogos.

1.2 Etapas en un proceso de visión artificial

Las etapas de un proceso de visión artificial se suelen clasificar en dos grupos (6). En un primer grupo se encuentran las etapas que ejecutan métodos de bajo nivel y, en un segundo grupo, las que realizan un procesamiento de la imagen de alto nivel o un análisis a nivel de escena. El objetivo de las etapas de bajo nivel es obtener las características más básicas de la imagen, como bordes, regiones y otros atributos simples. En el caso del procesamiento de alto nivel, se recogen las características extraídas en el nivel inferior y se construye una descripción de la escena.

A continuación, se describen de forma breve las etapas involucradas en este proceso. Dichas etapas son:

- **Adquisición de la imagen:** En esta etapa se captura una proyección en dos dimensiones de la luz reflejada por los objetos de la escena.
- **Pre procesamiento:** Se realizan tareas de eliminación de ruido y/o realce de la imagen.

- **Segmentación:** Detección de bordes y regiones que permite separar los diferentes elementos de la escena.
- **Extracción de características:** Se obtiene una representación formal de los elementos segmentados en la etapa anterior.
- **Reconocimiento y localización:** Mediante técnicas, como pueda ser la triangulación, se localiza al objeto en el espacio 3D.
- **Interpretación:** A partir de la información obtenida en las etapas previas y del conocimiento acerca del entorno se interpreta la escena.

1.3 Técnicas de adquisición de información 3D de una escena

La obtención de información 3D de una escena se puede realizar mediante técnicas muy diversas. Estas técnicas se dividen en dos grandes grupos como se muestra en la Figura 1. Las técnicas ópticas de visión artificial son un subconjunto dentro de las técnicas que se pueden utilizar para obtener información tridimensional de una escena. En la Figura 2 se muestra una clasificación de las diversas técnicas que pueden ser utilizadas en la adquisición de información 3D de una escena.

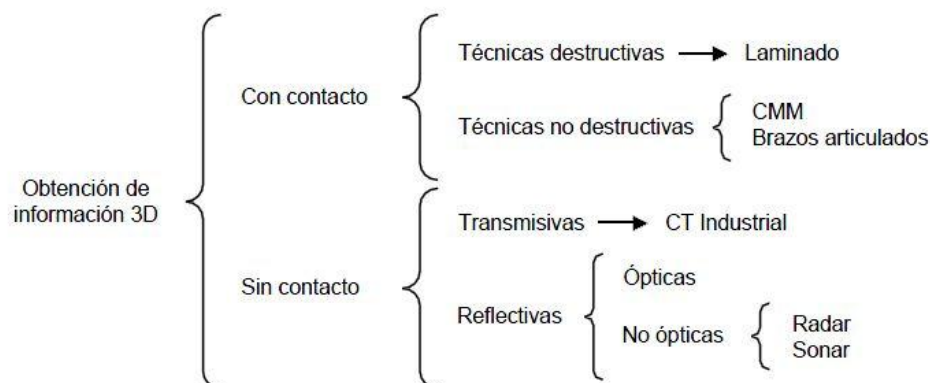


Figura 1. Clasificación de técnicas de adquisición de información 3D de una escena.

Las técnicas ópticas también se pueden clasificar en función de las características del sensor que escanea la escena.

- **Técnicas basadas en adquisición puntual:** Sólo es posible obtener un único punto de la escena en cada ciclo de muestreo.
- **Técnicas basadas en adquisición lineal:** Se obtienen una serie de puntos de la escena al mismo tiempo, correspondientes a una línea.
- **Técnicas basadas en adquisición matricial:** Proporcionan la forma más flexible de adquisición de una escena. La información obtenida se organiza en una cuadrícula o matriz (*grid*).
- **Técnicas basadas en múltiple vista:** Las coordenadas tridimensionales de un punto se obtienen a partir de múltiples imágenes de la escena que son capturadas simultáneamente.

Sobre este tema, se publicaron estudios comparativos de las principales técnicas de visión por computador para la obtención de información tridimensional (7). Estos trabajos se centran principalmente en la obtención de imágenes de rango utilizando técnicas ópticas activas.

El tipo de problema define qué técnicas deberían ser utilizadas para obtener unos resultados satisfactorios y cuáles deberían ser descartadas. En términos generales, esta selección se hace en base a criterios globales del problema, tales como el coste del sistema de visión y la disponibilidad de espacio para su ubicación, entre otros. Se definen varios parámetros objetivos como apoyo para la selección de la técnica más adecuada (8).

De entre estos parámetros los más destacables son:

- **Precisión:** indica la proximidad de la medida proporcionada por el sistema de visión al valor de la propiedad medida.
- **Resolución:** indica la magnitud más pequeña de la medida que puede detectar el sistema.
- **Sensibilidad:** indica el cambio de valor más pequeño que el sistema es capaz de detectar en la propiedad medida.
- **Incertidumbre:** define un rango centrado en la medida proporcionada por el sistema dentro del cual se encuentra, con una determinada probabilidad estadística, el valor real de la propiedad medida.
- **Error absoluto:** indica el valor absoluto de la diferencia entre la medida proporcionada por el sistema y el valor real de la propiedad medida.

En el caso de que el conjunto de parámetros indicado no sea suficiente para seleccionar la técnica óptica más adecuada para el problema en estudio, se pueden utilizar otros parámetros más específicos de los sistemas de visión, tales como campo de visión, profundidad de campo, tamaño de la imagen y frecuencia de muestreo.

1.3.1 Técnicas ópticas

Como se muestra en la Figura 2, las técnicas ópticas se suelen clasificar en dos grupos atendiendo a las características de las fuentes de luz utilizadas para iluminar la escena. En las técnicas pasivas no se tiene en cuenta la geometría de las fuentes de iluminación con respecto a la escena y al sensor, a la hora de calcular las coordenadas tridimensionales de los objetos, en cambio, en las técnicas activas se utilizan fuentes de luz con unas características muy específicas y su posición influye directamente a la hora de determinar dichas coordenadas. A esta fuente de iluminación se la suele denominar fuente de luz activa para distinguirla del resto de iluminación de la escena.

Técnicas pasivas

Las técnicas ópticas pasivas pueden ser aplicadas en un mayor número de situaciones que las técnicas ópticas activas. A continuación se presentan: la visión estéreo, la obtención de formas a partir de la sombra, a partir de la textura y a partir del movimiento, y la fotogrametría consideradas las principales técnicas ópticas pasivas (8).



Figura 2. Clasificación de técnicas ópticas.

Visión estéreo

Las técnicas de visión estéreo por computador analizan las diferencias de la proyección de la escena en dos imágenes tomadas desde dos posiciones diferentes como se muestra en la Figura 3. A las diferencias entre ambas imágenes se las denomina disparidad, y su análisis permite obtener la dimensión perdida en la proyección de la escena tridimensional en la imagen bidimensional (7). En el caso de los seres humanos el cerebro es el encargado de combinar ambas imágenes y de calcular su disparidad.

En el caso de un sistema de visión por computador, el análisis de las imágenes adquiridas simultáneamente por dos sensores distanciados espacialmente debe permitir establecer una correspondencia entre los puntos de ambas imágenes y realizar el cálculo de la profundidad a partir de las diferencias de posición de un punto entre las dos imágenes.

El cálculo de la correspondencia se realiza a partir de algoritmos que tratan de localizar las proyecciones de un punto de la escena en las dos imágenes capturadas. La principal limitación en este cálculo es que un único píxel de una imagen no proporciona información suficiente para identificar el punto correspondiente en la otra. Las soluciones que aplican estos algoritmos tratan de analizar los píxeles de su vecindad por medio de técnicas de correlación, detección y emparejamiento de bordes así como de segmentación de regiones, entre otras.



Figura 3. Técnica de visión estéreo. Imagen cortesía de la Universidad Politécnica de Oviedo.

El cálculo de la profundidad se basa en el conocimiento de la geometría del sistema de visión por computador, especialmente en la posición que ocupa cada una de las cámaras en el espacio. En el caso de que las dos cámaras estén perfectamente alineadas, configuración ideal del sistema, para calcular las coordenadas tridimensionales de un punto de la escena se aplican relaciones entre triángulos semejantes obtenidos de la geometría del sistema. En situaciones en que la geometría del sistema no cumpla la configuración ideal se debe modelar la transformación que liga los sistemas de coordenadas de ambas cámaras, o bien aplicar algún algoritmo de rectificación a las dos imágenes que proporcione las mismas imágenes en la configuración ideal.

Forma a partir de la sombra, la textura y el movimiento

En esta sección se describen algunas de las soluciones más utilizadas para la extracción de la forma de los objetos de una escena.

- Forma a partir de la sombra (*shape from shading, SFS*)
- Forma a partir de la textura (*shape from texture, SFT*)
- Forma a partir del movimiento (*shape from motion, SFM*)

En cualquier caso, suele ser frecuente el uso de estas técnicas de forma combinada para obtener información tridimensional acerca de los objetos de una escena. Uno de los ejemplos más recientes fue publicado en (28), en el que ellos reconstruyen la forma de superficies deformadas a partir de una sola imagen combinando técnicas de obtención de forma a partir de la sombra y la textura.

Se denomina fotometría a las técnicas que tratan de obtener información tridimensional de objetos a partir de las sombras que producen las fuentes de iluminación que inciden sobre sus superficies (8). Las técnicas que tratan de obtener información tridimensional de objetos a partir de la textura de sus superficies, se basan en dos efectos que se producen al observar un objeto en el que la textura de su superficie está descrita por un patrón regular. El análisis del ángulo y del tamaño relativo de esta variación en un sistema de visión por computador permite determinar la profundidad de cada uno de

los puntos de la escena a partir de la imagen bidimensional capturada.

Las técnicas de obtención de información tridimensional a partir del movimiento se basan en el análisis del movimiento relativo de los objetos de la escena con respecto al sensor del sistema de visión, así como en la traslación y rotación relativas de dichos objetos. A partir de este análisis se puede obtener la forma de los objetos y la profundidad de cada uno de los puntos de la escena; si el objeto a reconstruir es un cuerpo rígido, los puntos de su superficie que estén más próximos al sensor se moverán a una velocidad menor que los puntos que estén más alejados de éste. Dado que la velocidad será proporcional al movimiento del objeto y a la distancia de su superficie con el sensor, si se conoce con precisión el movimiento del objeto es posible estimar una imagen de rango a partir de cada uno de los puntos de la escena. Estas técnicas tienen la limitación de que no pueden ser aplicadas con objetos flexibles o que cambien de aspecto en el tiempo.

Fotogrametría

La fotogrametría es una disciplina que crea modelos en tres dimensiones (3D) a partir de imágenes dos dimensiones (2D), para de esta manera obtener características geométricas de los objetos que representan, mediante el uso de relaciones matemáticas establecidas en la geometría proyectiva, y de la visión estereoscópica que posee en forma natural el ser humano (9), donde las imágenes de los objetos son obtenidas por medios fotográficos como se muestra en la Figura 4, y la medición se realiza a distancia, sin que exista contacto físico con el objeto.

La primera utilización de la fotogrametría consistió en la realización de mapas y planos topográficos. De hecho, los mapas base de la cartografía de cualquier país, son obtenidos mediante ella. Actualmente, además de la realización de estos mapas base, se realizan muchos otros tipos de mapas de carácter especial, los cuales pueden presentar gran variedad de escalas, y se utilizan en el proyecto y diseño de obras tales como autopistas, carreteras, vías de ferrocarril, puentes, tuberías, oleoductos, gasoductos, líneas de transmisión, presas hidroeléctricas, estudios urbanos, entre otros.

Además de estos mapas, orientados principalmente al desarrollo de obras de ingeniería civil, se pueden mencionar mapas realizados para uso catastral, mapas geológicos, mapas de suelos y mapas forestales. Dentro de las disciplinas que se benefician de la fotogrametría no topográfica está la arquitectura, en el levantamiento de monumentos y de sitios; la arqueología, en aplicaciones similares a las usadas en arquitectura; la bioestereometría, en el estudio de formas de seres vivos; la construcción naval, la automotriz y la de maquinaria pesada hacen también uso de esta disciplina. (9).



Figura 4. Equipo fotográfico con dos cámaras para visión estéreo. Imagen cortesía de la Universidad Politécnica de Oviedo.

Técnicas activas

Las técnicas ópticas activas son consideradas las más interesantes desde el punto de vista de la reconstrucción tridimensional de una escena, dado que consiguen mayor precisión que las técnicas ópticas pasivas. Sin embargo, su implementación resulta mucho más costosa debido, entre otros factores, a las estrictas condiciones de iluminación necesarias.

Triangulación

La técnica de triangulación es la técnica más simple que se puede utilizar para obtener información tridimensional de una escena a partir de puntos fácilmente reconocibles en la misma. Para que ciertos puntos de la escena se puedan reconocer de forma sencilla, se suele proyectar sobre esta un patrón de luz estructurada. El análisis de la deformación que sufre el patrón al proyectarse sobre la superficie de los objetos de la escena permite extraer información tridimensional de estos. Los patrones que se utilizan en esta técnica son, entre otros, un punto o una línea generados a partir de una fuente de luz láser.

Esta técnica se basa en el cálculo de triángulos semejantes entre los elementos: sensor óptico, emisor de luz y objeto de la escena. El proceso de cálculo se puede realizar de múltiples formas atendiendo a las características y posición de cada uno de los elementos del sistema, aunque siempre se realiza estableciendo una relación de semejanza entre dos triángulos que comparten un vértice común: el punto focal de la cámara. Esta relación se puede observar en la Figura 5, donde el punto de la escena $P_w (X, Y, Z)$ se proyecta en la imagen como el punto $P_u (x, y)$.

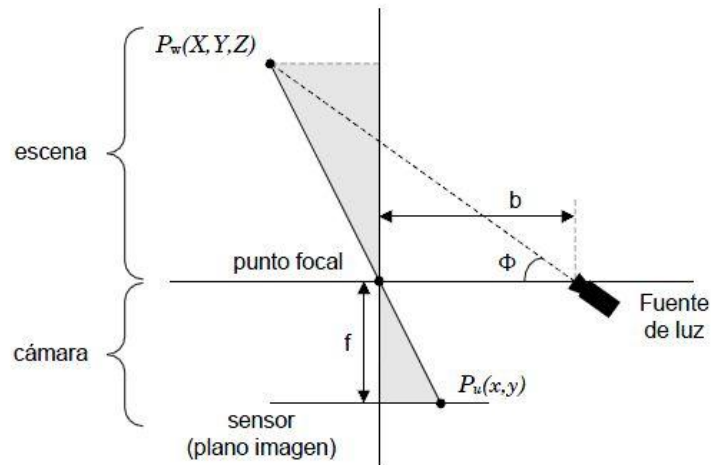


Figura 5. Relación de triángulos semejantes en la técnica de triangulación. Imagen cortesía de la Universidad Politécnica de Catalunya.

Luz estructurada

Las técnicas ópticas de luz estructurada, también denominadas de luz codificada, utilizan patrones de luz, proyectados sobre la escena para obtener información tridimensional de esta como se muestra en la Figura 6. En estas técnicas, la información tridimensional se obtiene analizando las deformaciones de la proyección del patrón sobre la escena con respecto al patrón original proyectado. El problema de la correspondencia entre los puntos de la imagen y los puntos del patrón original se resuelve de forma sencilla dado que los patrones de luz están codificados.

Existen diversas estrategias para codificar los patrones a proyectar sobre una escena (10): multiplexado temporal, vecindad espacial y codificación directa. Las estrategias de multiplexado temporal proyectan una secuencia de patrones diferentes sobre la escena a lo largo del tiempo, por lo que la estructura de cada patrón puede ser muy simple. En esta estrategia se suelen utilizar códigos binarios, códigos n-arios y códigos Gray. La codificación basada en la vecindad espacial crea un código para cada zona del patrón a proyectar a partir de los códigos asignados a sus vecinos. Los métodos de codificación más utilizados en este caso son secuencias de Bruijn y M-arrays. En el caso de la codificación directa, se define un código para cada píxel a partir de paletas de niveles de gris o de color.

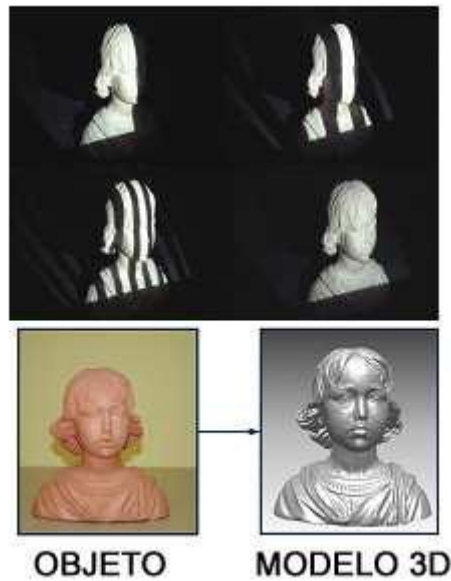


Figura 6. Modelado 3D con proyección de luz blanca estructurada. Imagen cortesía de la Universidad Politécnica de Oviedo.

Interferometría

Las técnicas interferométricas hacen interferir la luz emitida por los objetos de la escena con la generada por otras fuentes de iluminación. Por tanto, en la cámara del sistema de visión se obtiene un patrón de interferencia. La holografía convencional, la holografía conoscópica y las técnicas interferométricas basadas en láser y en el efecto Speckle son las más utilizadas actualmente. Cada uno de los píxeles de la cámara de un sistema de visión convencional registra información acerca de la intensidad de la luz que incide en él. En el caso de la holografía se busca, además de esta intensidad, la fase de la luz incidente.

Los rayos procedentes de un punto de la escena no son focalizados a un punto concreto de la imagen sino que llegan a todo el plano de esta, interfiriendo con los rayos de un haz de referencia, lo que genera un patrón de interferencia denominado holograma. Aunque con apariencia desordenada el holograma contiene información más completa de la escena que una fotografía ordinaria. Si los hologramas se generan a partir de luz incoherente, la técnica aplicada recibe el nombre de holografía conoscópica.

Finalmente, aunque no se trate en realidad de una técnica interferométrica, la técnica basada en patrones de Moiré suele clasificarse junto a estas debido a que las imágenes obtenidas y los algoritmos de procesamiento necesarios para obtener información tridimensional de la escena son similares. Matemáticamente, un patrón de Moiré es un patrón interferencial de baja frecuencia que se produce cuando dos patrones de mayor frecuencia espacial se interponen (8). El fenómeno de Moiré permite obtener información de todos los puntos de la escena a partir de una única imagen. Atendiendo a la forma en la que se genere el patrón de Moiré, se distinguen dos técnicas:

sombreado de Moiré y proyección de Moiré. En ocasiones también se utilizan variaciones o combinaciones de estas dos técnicas.

1.3.2 Tabla comparativa entre técnicas de adquisición tridimensional

Método	Ventajas	Inconvenientes
Técnica Activa: luz estructurada	<ul style="list-style-type: none"> • Exactitud • Precisión • Rapidez 	<ul style="list-style-type: none"> • Dependiente de la geometría del objeto • Ambientes controlados
Técnica Pasiva: fotogrametría	<ul style="list-style-type: none"> • Precisión • Reconstrucción Detallada • Robusto • Rapidez • Coste reducido 	<ul style="list-style-type: none"> • Geometría compleja • Problemas con ciertas texturas

Tabla 1. Tabla comparativa entre técnicas de adquisición tridimensional.

En la Tabla 1 se analizan dos técnicas de adquisición de información tridimensional para evaluar sus ventajas y retos para seleccionar una de ellas en aras resolver los objetivos planteados. Como se analiza en la tabla comparativa, existen diferencias que marcan el camino para la selección de una técnica de adquisición tridimensional que cumpla o resuelva los principales problemas detectados en la Línea de Imágenes y Videos basados en Rendering, donde se observó que la fotogrametría como técnica pasiva constituye una técnica que presenta ventajas ante la técnica de luz estructurada en cuestiones de coste reducido y la no dependencia de la geometría del objeto, las que marcan las diferencias entre ambas técnicas de adquisición.

1.4 Componentes de un sistema de visión por computador para la reconstrucción 3D

Un sistema de visión por computador para la reconstrucción de la forma tridimensional de una escena está compuesto por un conjunto de elementos. La distribución de estos elementos en el espacio, así como su distancia relativa con referencia a la escena, se denomina geometría del sistema de visión. Según las características de los elementos que forman el sistema de visión se establecerá una serie de restricciones que debe cumplir su geometría, definiendo un conjunto de configuraciones posibles. La elección de la geometría a implementar entre una de las posibles

configuraciones válidas se realiza en función de la naturaleza del problema y del entorno de instalación. En las siguientes secciones se describen los elementos principales que componen un sistema de visión por computador.

1.4.1 Fuente de iluminación

La iluminación es la característica que utilizan las técnicas de visión por computador para extraer información de los objetos de una escena. La iluminación de una escena se puede realizar con luz natural o mediante la utilización de lámparas. Las fuentes de iluminación más comúnmente utilizadas son lámparas halógenas, lámparas fluorescentes, proyectores, diodos LED y láser.

1.4.2 Cámara

La cámara de un sistema de visión por computador es el dispositivo que recibe la luz reflejada por la escena y la utiliza para generar imágenes. Uno de los elementos más utilizados de la cámara es su sensor ya que ese es el que determina la cantidad de mega píxeles que tendrá la cámara y está formado por una capa de material fotosensible que transforma la luz incidente en señales eléctricas. El sensor puede estar construido en base a una tecnología analógica (cámaras de tubo) o en base a una tecnología digital (cámaras de estado sólido).

1.4.3 Óptica

La óptica es uno de los elementos más importantes de cualquier sistema de visión. Su objetivo es concentrar la luz reflejada por los objetos de la escena en el sensor de la cámara para generar la imagen. La óptica está formada por un conjunto de lentes positivas o convergentes y negativas o divergentes, que aplican una transformación a la luz recibida de la escena, definida matemáticamente mediante una proyección perspectiva. Esta transformación consiste en una inversión y en un escalado de los objetos de la escena (8).

Todas las ópticas están definidas por un conjunto de parámetros. Una de las más utilizadas es la distancia focal de las lentes, es decir, la distancia que separa el centro de la lente de su foco. También se pueden destacar la distancia de enfoque y la profundidad de campo, que contribuyen a la nitidez con la que el sensor de la cámara generará la imagen. Además, se debe tener en cuenta que la transformación perspectiva que realiza la óptica no es perfecta, sino que introduce cierta distorsión que deforma la imagen con relación a la escena original. Esta distorsión tiende a ser mayor cuanto menor es la distancia focal, y afecta en gran medida a los sistemas de visión que requieran una precisión elevada, por lo que es necesario medirla y corregirla.

Las distorsiones que introduce una óptica, también denominadas aberraciones ópticas, modeladas matemáticamente según la ley de Snell, se deben a las propiedades de refracción de las lentes que

la componen. Aberraciones típicas son la distorsión radial, la distorsión descentrada o tangencial, la distorsión de prisma fino, la curvatura de campo, la aberración esférica y las aberraciones cromáticas, entre otras. La distorsión radial desplaza los puntos de la imagen radialmente a partir del centro, mientras que la distorsión tangencial los desplaza perpendicularmente a la línea radial. Las causas fundamentales que producen distorsiones en una óptica son un pulido defectuoso de la superficie de las lentes en el caso de la distorsión radial, una falta de alineación entre los componentes ópticos en el caso de la distorsión tangencial e imperfecciones en el diseño y ensamblaje de las lentes en el caso de la distorsión de prisma fino.

Las ópticas con lentes esféricas, que se han comentado hasta el momento, son las más comunes en cualquier sistema de visión. Sin embargo, en determinadas ocasiones este tipo de ópticas no representan la mejor opción para el sistema. Un ejemplo está en el problema de la diferencia entre la relación de aspecto del sensor de la cámara y la relación de aspecto del área de interés de la escena. Esta diferencia hace que se pierda cierto porcentaje de píxeles en las imágenes, por lo que no se aprovecha por completo la resolución del sensor de la cámara.

1.5 Calibración de un sistema de visión por computador

La óptica en un sistema de visión por computador aplica una transformación de perspectiva a la luz que incide sobre ella procedente de la escena. Esta transformación, denominada transformación directa, permite relacionar el sistema de coordenadas tridimensional de la escena con el sistema de coordenadas bidimensional de la imagen, dado que la transformación realizada sobre dicho sistema consiste en proyectar puntos del espacio tridimensional de la escena sobre el plano bidimensional de la imagen (11). En un sistema de visión por computador, la imagen generada por la cámara a partir de la luz procedente de la escena es sólo un medio para obtener información tridimensional de dicha escena. Para reconstruir tridimensionalmente la escena a partir de una imagen bidimensional es necesario realizar una transformación inversa que permita calcular las coordenadas tridimensionales correspondientes a cada punto de la imagen. Por tanto, es necesario conocer con exactitud la transformación que realiza la cámara para poder construir un modelo matemático que permita realizar la operación inversa.

La calibración es el proceso que permite determinar los parámetros de la transformación entre puntos de la escena y puntos de la imagen, es decir, de la transformación perspectiva. Esta transformación está definida por dos tipos de parámetros:

Parámetros intrínsecos. Caracterizan las propiedades físicas de una cámara y de su óptica. Habitualmente se suelen considerar los siguientes:

- distancia focal: f

- desplazamiento del centro de la imagen: (C_x, C_y)
- coeficiente de distorsión radial: K_1, K_2

En ocasiones estos parámetros son suministrados directamente por el fabricante de la cámara, aunque existen multitud de técnicas para su obtención.

Parámetros extrínsecos. Definen la posición y orientación de la cámara con respecto al sistema de coordenadas de la escena:

- Traslación: T_x, T_y, T_z
- Rotación: α, β, γ

Existen muchas técnicas para obtener los parámetros extrínsecos de una cámara, aunque los más habituales son los que utilizan conocimiento específico acerca de la escena. En la Figura 7 se muestran algunas de las representaciones de los parámetros de la cámara más importantes que intervienen en el proceso de adquisición tridimensional.

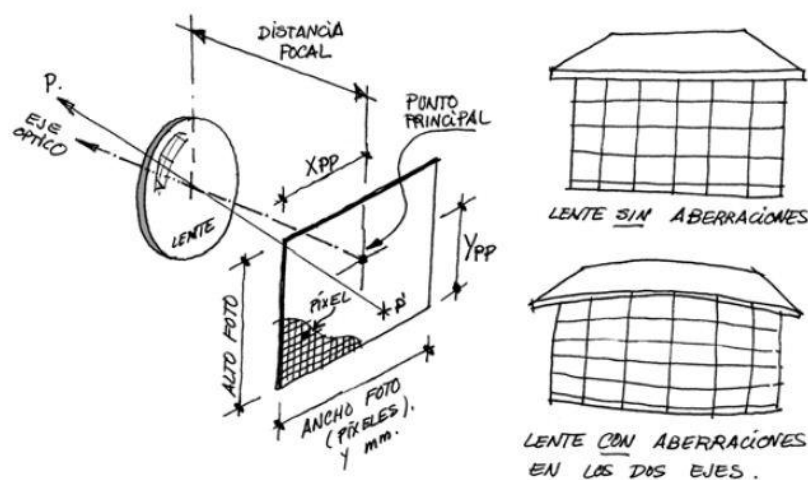


Figura 7. Parámetros de calibración de una cámara. Imagen cortesía de la Universidad Politécnica de Madrid.

La calibración de una cámara se divide en dos fases. En la primera de ellas se modela matemáticamente el comportamiento físico y óptico del sensor, definiendo un conjunto de parámetros intrínsecos y extrínsecos del modelo. En la segunda fase se estiman los valores de los parámetros del modelo obtenido en la fase anterior. Esta segunda fase se divide a su vez en el siguiente conjunto de tareas:

1. Situar un conjunto de referencias en la escena y capturar una serie de imágenes de la misma para determinar con precisión la posición de un conjunto de puntos tridimensionales.
2. Determinar la correspondencia entre los puntos proyectados en la imagen y la referencia correspondiente en la escena.

3. Ajustar los parámetros del modelo que resuelve la correspondencia entre puntos en la imagen y referencias en la escena.

Un método de calibración debe cumplir las siguientes características (12):

- ✓ **Autonomía:** No debe requerir intervención por parte del usuario del sistema de visión por computador.
- ✓ **Precisión:** Debe lograr precisión requerida por el sistema.
- ✓ **Eficacia:** El procedimiento completo de calibración debe tener un coste computacional asumible por el sistema.
- ✓ **Versatilidad:** Debe poder calibrar el sistema aun con diferentes ópticas y distintos niveles de precisión.

1.6 Formatos de ficheros para la obtención de nube de puntos

Existen diferentes formatos de ficheros en los que se pueden almacenar información tridimensional y utilizar en diferentes tareas en el área del modelado de objetos. A continuación se analizan algunos ficheros para seleccionar los que mejores se adapten para la construcción de la solución.

1.6.1 Formato de fichero OBJ

Los archivos OBJ definen la geometría y otras propiedades de los objetos de visualización de avanzada (17). Los archivos de objetos también se pueden utilizar para transferencias de datos geométricos entre el visualizador y otras aplicaciones. El formato es compatible con objetos poligonales y objetos de forma libre, la geometría poligonal utiliza puntos, líneas y caras para definir los objetos mientras que la geometría de forma libre utiliza curvas y superficies.

1.6.2 Formato de fichero TXT

Los archivos de texto (.txt) sólo pueden contener recursos de cadena (17). Cuando se guarda correctamente un archivo de texto, se pueden especificar cadenas mediante uno de los tres esquemas de codificación. Si se desea guardar y utilizar caracteres de datos tridimensionales, se guarda el archivo de texto mediante la codificación UTF-8 en vez de usar una página de código ANSI específica.

1.6.3 Formato de fichero VRML

El formato VRML (Virtual Reality Modeling Lenguaje, “Lenguaje de Modelado para Realidad Virtual”) se creó con el objetivo de realizar escenas y modelos vectoriales en 3D con las que el usuario pudiera tener cierta interacción y pudieran ser cargados en la web. En el archivo se especifican las coordenadas de los vértices que componen el elemento, colores, hipervínculos, luces, asignación de mapas, propiedades de la escena, etc. Los archivos VRML a menudo se conocen como mundos y tienen una extensión .wrl, muchos programas en 3d pueden guardar en este formato.

1.6.4 Formato de fichero PLY

El formato PLY es una descripción simple de objetos que fue diseñado como un formato conveniente para los investigadores que trabajan con modelos poligonales. El formato fue diseñado principalmente para almacenar datos de tres dimensiones a partir de escáneres 3D. Es compatible con una descripción relativamente simple de un solo objeto como una lista de polígonos nominalmente plana (27). Una variedad de propiedades se pueden almacenar como: color y transparencia, superficies normales, coordenadas de textura y valores de datos de confianza. El formato permite tener propiedades diferentes para el frente y la espalda de un polígono.

CAPÍTULO 2. Solución Propuesta

Introducción

Después del estudio teórico realizado anteriormente, en este capítulo se describe la técnica y algoritmos básicos que conformaron la solución del proceso de adquisición tridimensional de un objeto del mundo real. Se explican además las etapas en el proceso de adquisición tridimensional según la técnica seleccionada, como punto fundamental en la investigación realizada para la construcción de la solución y se mencionan los recursos a utilizar en el proceso de desarrollo.

2.1 Selección de la técnica de adquisición geométrica

La fotografía en su sentido más amplio es un proceso que convierte el mundo real de tres dimensiones en imágenes planas de dos dimensiones como se muestra en la Figura 8. La cámara es el dispositivo que hace esta transformación o dicho en otra forma, traza un mapa de tres dimensiones a dos dimensiones, no se puede trazar un mapa del mundo de tres dimensiones en dos dimensiones completamente, siempre alguna información se pierde, principalmente la profundidad.

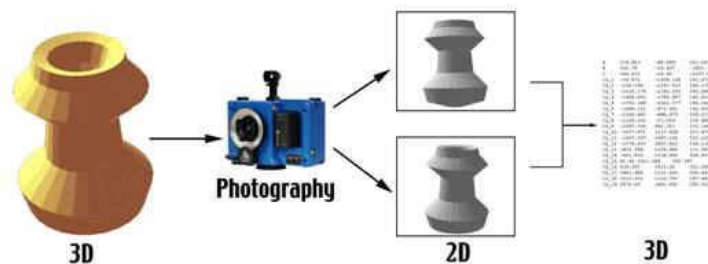


Figura 8. Proceso fotográfico de la adquisición tridimensional.

Lamentablemente, la fotografía y el proceso de medida no son perfectos y la reconstrucción 3D es también imperfecta, sin embargo, se pueden tomar más fotografías y usar la información suplementaria en ellas para mejorar el proceso. Las coordenadas de tres dimensiones que se producen a partir de las medidas de múltiples fotografías son el resultado final de la **fotogrametría**, la cual constituye la técnica seleccionada para el proceso de adquisición que se desarrolló como resultado de la investigación, y su elección se contempló en función de sus aplicaciones, y de un estudio de las ventajas y retos en el área de la visión por computadora.

Cuando se ven los objetos en relieve es debido a que los ojos proporcionan al mismo tiempo, dos visuales del mismo objeto desde dos puntos de vista ligeramente distintos que intersectan, estas dos imágenes son mezcladas en el cerebro y como consecuencia se puede ver una tercera dimensión.

Este principio de **estereoscopia** es aplicado potentemente en la fotogrametría, para poder extraer la tercera dimensión a partir de imágenes bidimensionales. En realidad, lo que se hace en una reconstrucción fotogramétrica es sustituir nuestros ojos por la cámara, y la distancia inter-pupilar por la distancia entre disparos consecutivos. La incorporación de una segunda perspectiva de la misma zona incrementa notablemente la información tridimensional, con la incorporación del concepto de visión estéreo o estereoscopia. Para restituir un modelo 3D a partir de pares de fotografías estereoscópicas se investigaron los mismos principios utilizados por nuestro propio sistema de visión, donde se pudo concluir que para que se puedan reproducir la estereoscopia se deben dar dos condiciones esenciales: que cada ojo vea sólo la perspectiva que le corresponde y que las visuales tengan intersección entre sí. Con respecto a este último aspecto, esa intersección se produce cuando los fotogramas tienen zonas en común, por lo que ya se sabe por qué es tan necesario el recubrimiento estereoscópico (23).

2.2 Etapas en el proceso de adquisición geométrica

Cuando se trabaja con la fotogrametría se trabaja con la adquisición de imágenes y con la modelación de las cámaras. Para una descomposición en mayor detalle del proceso de adquisición de información tridimensional mediante esta técnica, se determinaron seis pasos fundamentales:

1. **Adquisición de pares estéreos:** Se obtienen las imágenes estéreo del patrón para la calibración y del modelo para adquirir su información tridimensional.
2. **Modelado de la cámara (geometría del sistema):** Un modelo de cámara es una representación de los atributos geométricos y físicos más importantes de las cámaras utilizadas para la visión estéreo.
3. **Obtención de puntos de control:** Emparejar en las diferentes imágenes aquellos puntos 2D procedentes de un punto 3D común.
4. **Rectificado de imágenes:** Transformar mediante las imágenes obtenidas, conociendo los parámetros tanto intrínsecos como extrínsecos de las cámaras, de manera que las líneas epipolares de ambas imágenes queden alineadas horizontalmente.
5. **Calculo de disparidad o profundidad:** Determinar mediante la medición de la disparidad de dos elementos semejantes en dos imágenes estéreos, la profundidad de los puntos del objeto en escena, recuperando la tercera dimensión perdida durante el proceso de adquisición de la imagen.
6. **Adquisición de nube de puntos:** Se obtiene la nube de puntos tridimensionales de la escena fotografiada.

2.3 Adquisición de imágenes

La adquisición de imágenes, tanto los pares estéreos de calibración como los pares estéreos del objeto a restituir, constituye una etapa fundamental en el proceso de obtención de la nube de puntos y su análisis organiza series de aspectos determinantes en la calidad del resultado final, los cuales serán tratados a continuación.

Pares estereoscópicos del patrón de calibración

Para obtener los parámetros de calibración será necesario obtener unas 20 fotografías a un patrón de calibración como se muestra en la Figura 9, donde el patrón deberá aparecer perfectamente enfocado de forma que sean visibles todas las esquinas de la cuadrícula y ajustados para que el patrón abarque la mayor parte de la imagen posible, siempre dejando espacio para el reborde blanco del patrón.

El patrón debe fotografiarse de forma que la perspectiva de las cuadrículas fugue en todas la direcciones posibles (hacia la izquierda abajo, izquierda arriba, derecha arriba, de abajo a arriba, etc.). Todo esto es importante porque se utilizarán algoritmos de detección de esquinas y cuanto más información tenga, menor error cometerá en la calibración. Es importante también que todas las fotografías tengan condiciones lumínicas lo más parecidas posibles.



Figura 9. Ejemplos de patrones de calibración.

Par estereoscópico del objeto a restituir

Para la restitución del modelo 3D de un objeto se realizan pares estereoscópicos de fotografías. La calidad de las fotografías que se tomen y el control de los errores que se cometan serán directamente proporcionales al éxito de la adquisición tridimensional. Por lo tanto es muy importante conocer la manera correcta de tomar las fotografías y controlar todos los parámetros en el proceso de adquisición.

Colocación de la cámara

Una vez colocado todo el sistema y realizado las fotografías, se tendrá en cuenta la colocación de la cámara en una posición que venga bien y teniendo en cuenta que jamás se modificará el zoom de la cámara, ya que se estaría anulando la calibración que se habrá realizado. La cámara se coloca en un trípode y a la hora de realizar las fotografías se trata de evitar cualquier contacto físico para asegurar que la cámara esté bien estable.

Movimiento de la cámara

Cuando se realice las dos fotografías se intentará que transcurra el mínimo tiempo posible entre ambas tomas. Se deberá tener en cuenta que para que la restitución tenga éxito el solapamiento de las imágenes debe ser por lo menos de un 70-80%. El movimiento de la cámara se llevará a cabo de forma que el eje de la cámara este paralelo entre una posición y la otra, apareciendo solo una disparidad horizontal entre ambas imágenes. Una forma de asegurarse es colocar marcas, para mover el trípode. A pesar de las precauciones que se deberán tomar, es inevitable cometer pequeños errores en la posición de la cámara, estos errores se verán más adelante cuando se vea la rotación y translación que aparece al calibrar extrínsecamente la cámara.

2.4 Modelo de cámaras. Geometría 3D

El sistema de visión humano puede modelarse mediante un par de cámaras que representan cada uno de los ojos. Cada cámara está definida por un conjunto de parámetros que identifican su posición en el espacio tridimensional y sus posibles distorsiones. Para llevar a cabo la proyección de un punto 3D en una imagen 2D, se trabaja con el modelo de cámaras "Pinhole".

La imagen obtenida por una cámara fotográfica es muy similar a la que obtiene el ojo humano. El modelo matemático y geométrico que mejor explica su formación es el de la perspectiva cónica, modelo "Pinhole" (14). La gran mayoría de procedimientos de calibración se basan en el modelo de cámara Pinhole. Este modelo está basado en la geometría proyectiva, la proyección de un punto de la escena se obtiene de la intersección de una línea que pasa por ese punto y el centro de proyección (plano foco) con el plano imagen, como se muestra en la Figura 10.

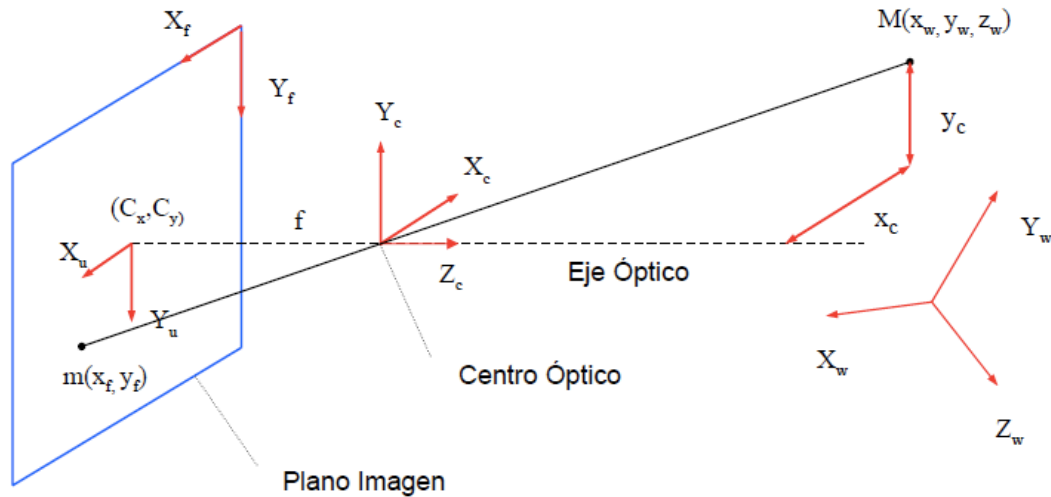


Figura 10. Geometría proyectiva. Imagen cortesía de la Universidad Politécnica de Oviedo.

El modelo Pinhole es la base teórica de la calibración intrínseca pues en él se explican los dos conceptos fundamentales de dicha calibración, como se muestran en la Figura 11 y que se señalan a continuación:

- ✓ Matriz de calibración intrínseca.
- ✓ Distorsión radial.

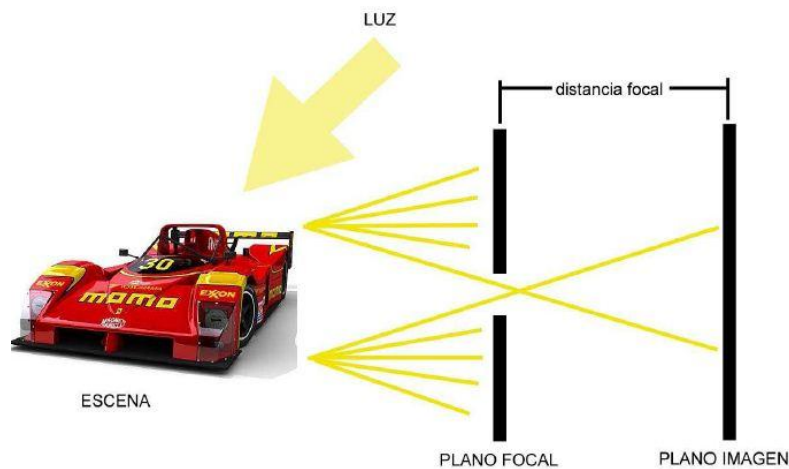


Figura 11. Modelo ideal de cámara, Pinhole. Imagen cortesía de la Universidad Politécnica de Oviedo.

Dentro de la investigación realizada se estudió la óptica como uno de los componentes más importantes dentro de un sistema de adquisición tridimensional y es aquí donde se demuestra su relevancia en aras de entender dicho modelo. La información 3D del mundo se proyecta en el plano de la imagen a través del centro óptico situado a una distancia igual a la distancia focal f del plano de la imagen.

La característica principal de las imágenes obtenidas mediante el modelo de cámara Pinhole es que cuentan con una profundidad de campo infinita. Esto implica que en la imagen fotográfica

resultante se puede ver claramente cualquier punto sin necesidad de realizar un encuadre especial, como debería hacerse al utilizar cualquier lente de un equipo profesional. Cada punto refleja la luz que recibe en forma de un haz que se extiende en todas las direcciones. Lo que realiza el modelo Pinhole es seleccionar un único rayo del haz, desechando el resto, de forma que solo uno de los rayos llega a la superficie sensible, formando un punto de la imagen.

2.4.1 Interpretación geométrica y matricial del modelo Pinhole

Para entender un poco más este modelo se realizó un análisis geométrico y matricial. En el modelo Pinhole el observador es reducido a un punto de coordenadas XYZ. En una determinada posición del espacio se encuentra el plano de proyección, que representa el negativo de la cámara fotográfica (o la retina en el ojo). En la Figura 12 se observa la proyección A' de un punto A del espacio, obtenido por la intersección del plano del sensor con la recta que une el vértice de proyección con ese punto.

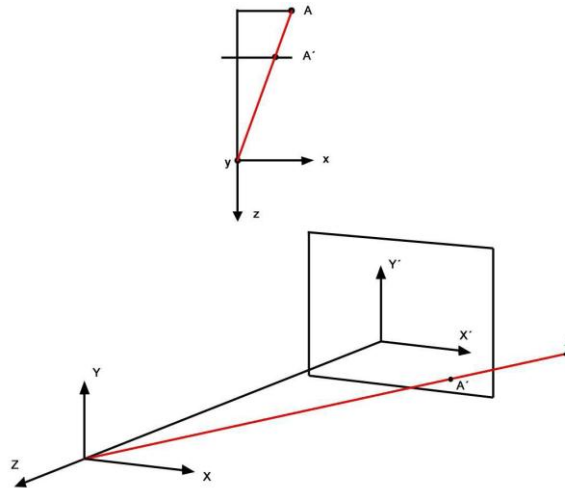


Figura 12. Proyección de un punto A en el espacio. Imagen cortesía de la Universidad Politécnica de Oviedo.

Si se hace coincidir el origen de coordenadas con la posición del observador y el plano de proyección se centra en la prolongación a la perpendicular del eje $-Z$, estando los ejes X e Y definiendo la horizontal y vertical respectivamente, un punto $P(X, Y, Z)$ se proyecta en un punto $P'(x', y')$ de acuerdo a las siguientes expresiones:

$$\begin{aligned} x' &= f \cdot \frac{X}{Z} \\ y' &= f \cdot \frac{Y}{Z} \end{aligned} \tag{2.1}$$

Donde f es la distancia focal de la cámara. Estas ecuaciones se representan de forma matricial de la forma:

$$Z \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.2)$$

Tanto las coordenadas de P como de P' vienen dadas en las mismas unidades, por ejemplo mm, sin embargo cuando se mide sobre una fotografía digital las unidades serán en píxeles. Los píxeles del sensor de la cámara tienen unas dimensiones físicas, no necesariamente tienen porque tener el mismo alto que ancho (no siempre son perfectamente cuadrados), por lo que las coordenadas del punto proyectado P'' (x'', y''), en píxeles, se obtienen aplicando los factores de escala Sx y Sy (píxel/mm):

$$\begin{aligned} x'' &= Sx \cdot x' \\ y'' &= Sy \cdot y' \end{aligned} \quad (2.3)$$

Además, habitualmente los píxeles de una foto se leen desde la esquina superior izquierda, de izquierda a derecha y de arriba abajo; y teniendo en cuenta que el punto medio de la foto no tiene por qué coincidir con el centro de proyección (el sensor puede estar desplazado debido a errores de montaje) se tiene la siguiente expresión del P''' (x''', y''') expresado en píxeles respecto del sistema de coordenadas centrado en la esquina superior izquierda:

$$\begin{aligned} x''' &= x'' + C_x = Sx \cdot x' + C_x \\ y''' &= y'' + C_y = Sy \cdot y' + C_y \end{aligned} \quad (2.4)$$

En forma matricial se tiene que:

$$\begin{aligned} Z \begin{pmatrix} x''' \\ y''' \\ z''' \end{pmatrix} &= \begin{pmatrix} S_x & S_\varphi & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} f S_x & S_\varphi & 0 \\ 0 & f S_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \end{aligned} \quad (2.5)$$

Donde se ha introducido el término S_φ para reflejar que los píxeles no tienen porque ser rectangulares, sino trapezoidales.

2.4.2 Distorsiones geométricas

Habitualmente es necesario considerar defectos en el sistema de lentes que hacen que este modelo no sea exacto. Las distorsiones geométricas vienen dadas por la distancia existente entre la posición ideal dada por el modelo Pinhole y la posición real que el píxel correspondiente en la imagen (2). Existen dos tipos de distorsiones geométricas, la tangencial, que aparece debido a la

no perpendicularidad del eje óptico de la lente fija y el plano imagen, y la radial, con mayor relevancia dentro de las etapas del proceso de adquisición y que se explica a continuación.

Distorsión radial

La distorsión radial no puede ser ignorada especialmente para distancias focales cortas. En el modelo Pinhole se supone que todos los rayos provenientes de un cierto objetivo atraviesan un cierto agujero para impactar en el sensor imagen. Sin embargo las lentes no tienen este comportamiento lineal, por lo que el modelo Pinhole debe ser corregido con los parámetros de distorsión radial para que el comportamiento del modelo se asemeje a la realidad.



Figura 13. Corrección de la distorsión radial. Imagen cortesía de la Universidad Politécnica de Oviedo.

La distorsión radial hace que los píxeles se alejen más del centro de la foto cuanto más lejos estén. Como se observa en la Figura 13, las líneas rectas aparecen representadas como curvas debido a la distorsión radial. Matemáticamente la modelización se lleva a cabo mediante las siguientes ecuaciones que relacionan $P(x_p, y_p)$, que es el punto en coordenadas píxel de la foto original, con $P(x''', y''')$, que es el mismo punto en la foto con la distorsión radial corregida. Al tratarse de una transformación no lineal no puede ser integrada en las expresiones matriciales previas:

$$x''' = x_p \cdot (1 + a \cdot r^2)$$

$$y''' = y_p \cdot (1 + a \cdot r^2)$$

Con:

$$r^2 = x_p^2 + y_p^2$$

$$a = cte. \tag{2.6}$$

Geoméricamente la distorsión radial se puede interpretar como indica la Figura 14, el punto no se debería proyectar sobre el plano del negativo sino sobre una superficie curva, por lo que cuanto más alejado esté x''' del origen de coordenadas mayor será el error debido a la distorsión radial (15).

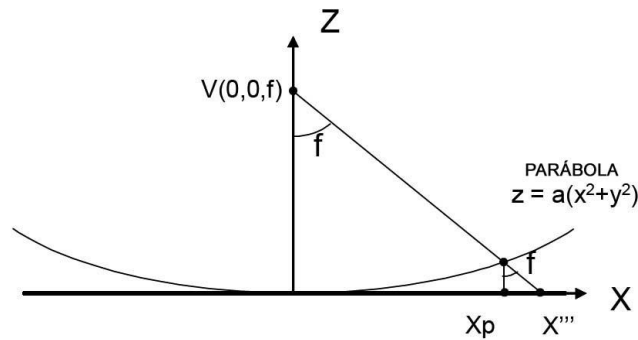


Figura 14. Parábola del negativo. Imagen cortesía de la Universidad Politécnica de Oviedo.

2.4.3 Matriz intrínseca

Con la calibración intrínseca se obtiene información sobre la distorsión radial y la matriz intrínseca, cuya expresión es:

$$\begin{pmatrix} fx & 0 & Cx \\ 0 & fy & Cy \\ 0 & 0 & 1 \end{pmatrix}$$

Donde:

- ✓ f: focal de la cámara
- ✓ Cx, Cy: mitad del tamaño del negativo (en píxel), determinan el centro.

Se observa que esta matriz coincide con una de las matrices obtenidas en el desarrollo de Pinhole suponiendo que los píxeles son rectos. Es decir, la base teórica de la calibración intrínseca es el modelo de Pinhole y en la que se podrá extraer informaciones necesarias para obtener un par de imágenes bien calibradas con el fin de obtener una buena adquisición tridimensional.

2.5 Obtención de puntos de control entre dos cámaras que ven una misma escena

Los sistemas de referencia de las cámaras izquierda y derecha están relacionados a través de los parámetros extrínsecos. Cada cámara identifica un sistema de referencia 3D, el origen del cual se sitúa en el centro de proyección de la cámara y el eje Z con el eje óptico. Los parámetros extrínsecos definen una transformación rígida en el espacio tridimensional definida por un vector de translación y uno de rotación.

2.5.1 Geometría epipolar

La relación geométrica entre dos proyecciones de un mismo punto físico puede ser expresada por medio de la geometría epipolar. Cuando dos cámaras captan una escena 3D desde dos puntos distintos, hay una serie de relaciones geométricas entre los puntos tridimensionales de la escena y sus proyecciones en las imágenes 2D que llevan a una correlación entre los puntos de la imagen.

Estas relaciones son derivadas basándose en la hipótesis de que las cámaras pueden ser aproximadas a un modelo de cámara ideal Pinhole.

En la Figura 15 se podrá ver un esquema de dicha geometría, donde:

- ✓ C_L, C_R = punto donde se sitúa el centro de proyección de la cámara izquierda (L, left) y la cámara derecha (R, right).
- ✓ X = punto de la escena observado por ambas cámaras.
- ✓ x, x' = proyecciones del punto X en el plano de proyección de cada cámara.

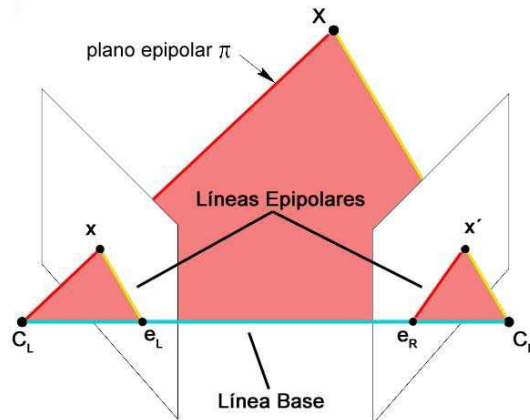


Figura 15. Geometría epipolar 1. Imagen cortesía de la Universidad Politécnica de Oviedo.

La geometría epipolar recibe el nombre de los epipolos, que son los puntos en los cuales la recta que une los centros de proyección de las cámaras corta a los planos de proyección.

- ✓ e_L = epipolo izquierdo.
- ✓ e_R = epipolo derecho.

Por construcción geométrica, los epipolos representan la proyección en su correspondiente plano del centro de proyección de la otra cámara. En el caso de que uno de los planos imagen sea paralelo a la recta que une los centros de proyección, el epipolo de este plano estaría situado en el infinito. La línea de base conecta los centros ópticos C_L y C_R y corta a los planos de proyección de las cámaras en los epipolos. El plano formado por los centros de proyección y el punto X observado en la escena se denomina plano epipolar. Este plano se intersecta con los planos de la imagen en las denominadas líneas epipolares, las cuales a su vez cortan a la línea base en los epipolos.

Las líneas epipolares constituyen la proyección en la imagen izquierda de la línea que une el centro óptico de proyección de la cámara derecha con el punto X de la escena y viceversa. Es importante entender la geometría epipolar porque simplifica la búsqueda de un mismo punto en dos imágenes diferentes, es decir, la búsqueda del mismo píxel en dos fotografías. En la Figura 15 se puede ver claramente que las proyecciones x y x' quedan comprendidas en las líneas epipolares, lo que reduce el problema a una única dimensión.

2.5.2 Correspondencia de un punto en dos imágenes.

Para la construcción del sistema de adquisición de información tridimensional será necesario analizar como se podría encontrar el punto de la imagen izquierda en la imagen derecha, procedente de un punto 3D común. La matriz fundamental F permite hallar la recta epipolar en una imagen conocido un punto en la otra imagen. Esta matriz es constante para una geometría bifocal dada. Por tanto conocido un punto en la imagen izquierda se obtiene la recta epipolar en la imagen derecha, de tal forma que se reduce significativamente la zona de búsqueda del punto en la imagen derecha. Sin embargo aún hay que realizar una búsqueda por la línea epipolar.

Existen varios métodos para obtener la correspondencia, estos métodos se pueden dividir en locales y globales (2). Debido a que generalmente se obtienen falsas correspondencias, sobre todo en zonas donde las intensidades del punto son muy parecidas, o cuando los puntos en la imagen izquierda no aparecen en la imagen derecha y viceversa, se determinó realizar la correspondencia estéreo o la relación existente entre dos cámaras, manualmente. Cuando se parte de la correspondencia de puntos de dos imágenes, comúnmente se emplea la matriz fundamental para obtener los parámetros de la cámara. Es posible estimar la matriz fundamental a partir de más de siete puntos de control, al usar esta matriz fundamental se puede obtener la matriz de proyección y realizar la proyección para la adquisición de información tridimensional en un espacio proyectivo. La solución se construirá en base a la correspondencia manual para la extracción de los parámetros extrínsecos de la cámara, tanto el vector de traslación como el de rotación. En la Figura 16 se muestra la correspondencia de puntos en dos vistas procedentes de un punto 3D común.

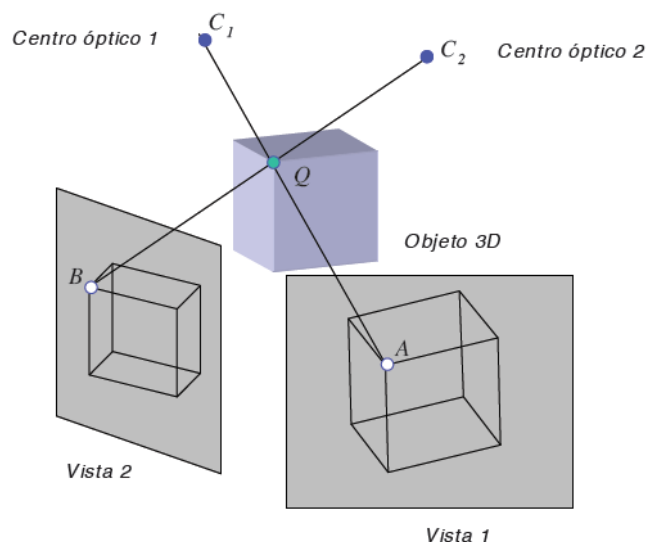


Figura 16. Correspondencia de un punto en dos imágenes procedentes de un punto 3D común. Imagen cortesía de la Universidad Politécnica de Oviedo.

2.5.3 Parámetros extrínsecos

Los parámetros extrínsecos definen la relación espacial entre la cámara y el mundo real. Es decir la orientación y la rotación respecto a un sistema de coordenadas conocido, habitualmente denominado como sistema de coordenadas del mundo real.

Desplazamiento.

Relacionan la posición de la cámara respecto al sistema de coordenadas del mundo real, que vendría dado por un vector (TX, TY, TZ) .

Orientación.

Relacionan la orientación de la cámara respecto al sistema de coordenadas del mundo real. Vector (RX, RY, RZ) que indica los ángulos respecto a los ejes X, Y, Z.

2.6 Rectificación de imágenes

Cuando se realizan las fotografías aparecen giros y movimientos de alguna de las cámaras que impiden que las imágenes estén alineadas (es decir, el mismo punto en las dos fotografías aparece desplazado). Para solucionar este problema es necesario recurrir a la rectificación, que se basa en la geometría epipolar. Para llevarla a cabo es necesario conocer tanto los parámetros intrínsecos como los parámetros extrínsecos de las cámaras.

Básicamente rectificar significa transformar mediante software las imágenes obtenidas, conociendo los parámetros tanto intrínsecos como extrínsecos de las cámaras, de manera que las líneas epipolares de ambas imágenes queden alineadas horizontalmente (4). Con esto se consigue que los puntos p_l y p_r tengan la misma posición en el eje "y" en sus respectivas imágenes. En la Figura 17, se ve claramente el objetivo de la rectificación.

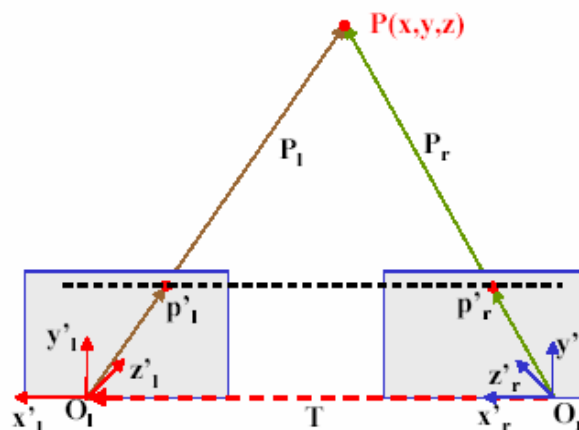


Figura 17. Planos de imagen alineados. Imagen cortesía de la Universidad Politécnica de Oviedo.

Para conseguir normalizar, es necesario alinear los ejes ópticos de manera que los planos de imagen transformados queden paralelos entre sí. Se seguirán 3 pasos fundamentales para la rectificación:

1. Rotar la cámara derecha e izquierda, hasta que sus ejes "x" coincidan: $O_r - O_l = T$
2. Definir la matriz de rotación R_{rect} para la nueva posición.
3. Por último calcular las nuevas matrices de proyección y reprojectar los puntos 3D sobre los nuevos planos de imagen.

En la Figura 18, se puede ver el proceso de normalización de un par estereoscópico, donde se consigue que las líneas epipolares de ambas imágenes queden alineadas horizontalmente.

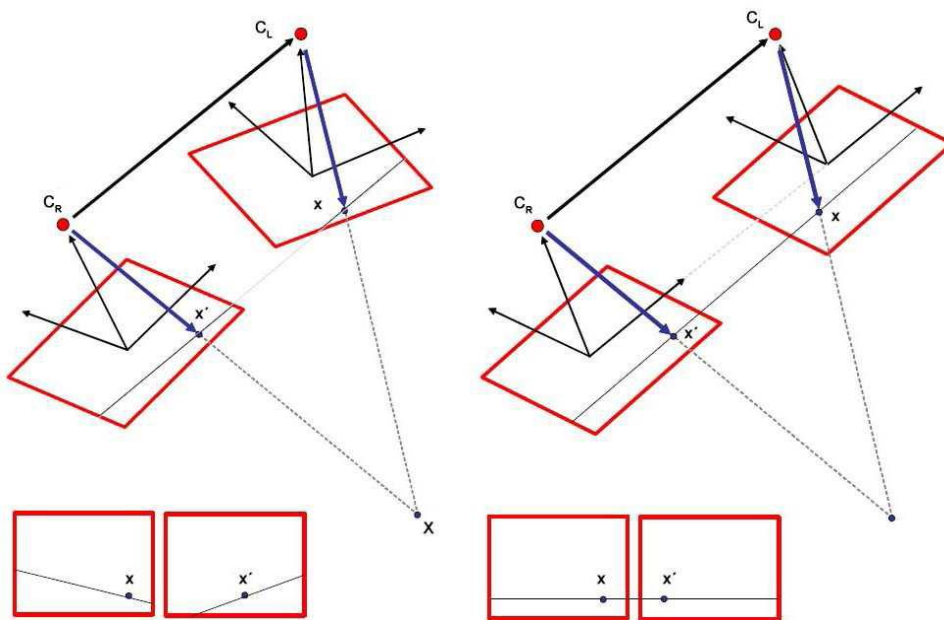


Figura 18. Normalización de un par estereoscópico. Imagen cortesía de la Universidad Politécnica de Oviedo.

2.7 Cálculo de disparidad o profundidad

Los sistemas de visión estereoscópica permiten obtener distancias de objetos dentro de una escena a partir de un par de imágenes de la misma. Las distancias se obtienen a partir de la medición de la disparidad de dos elementos semejantes en ambas imágenes. Las disparidades de todos los puntos de una imagen conforman el denominado mapa de disparidad como se muestra en la Figura 19.

Si el par de imágenes estéreo ha sido previamente rectificado, las líneas epipolares conjugadas se vuelven colineales y paralelas a la línea base del sistema estéreo. Esto significa que puntos que se corresponden en ambas imágenes tienen la misma coordenada en el eje vertical y, reduciendo el

problema de correspondencia a la búsqueda en una sola dimensión (eje horizontal x). La principal guía para resolver el “apareamiento” de un par de imágenes estéreo (problema de correspondencia), se basa en la semejanza o la no semejanza. Características o áreas correspondientes deben ser similares en ambas imágenes del par estéreo. Existen diferentes medidas de semejanza, difieren entre sí en el costo computacional, la eficiencia, etc. Básicamente se pueden dividir las diversas técnicas en dos grupos: las basadas en características, y las basadas en la correlación. Los métodos basados en características restringen la búsqueda a un conjunto disperso de características. Emplean propiedades simbólicas y numéricas de las características, disponibles a partir de los descriptores de características. En los métodos basados en la correlación, los elementos a comparar son ventanas de la imagen de dimensión fija, y el criterio de semejanza es una medida de la correspondencia entre las ventanas de las dos imágenes. El elemento correspondiente queda determinado por la ventana que maximiza el criterio de semejanza dentro de la región de búsqueda.

Dentro de este segundo grupo se pueden distinguir tres métodos para el cálculo del mapa de disparidad: la suma de diferencias absolutas (sum of absolute differences: SAD), la suma de diferencias al cuadrado (sum of squared differences: SSD), y la correlación cruzada normalizada (normalized cross correlation: NCC) (18). A partir del análisis de pares estereoscópicos de diferentes bibliografías, se observó que no existen grandes diferencias respecto a la calidad de los mapas de disparidades obtenidos por los diferentes métodos evaluados. Sin embargo, dentro de los métodos basados en correlación, SAD es el que mejor desempeño presenta respecto al costo computacional (26), por lo que se utilizará dicho método para la obtención final de la nube de puntos y su elección se contempló en función de su aplicación y que constituye una funcionalidad de la biblioteca que se utilizará en la construcción de la solución.

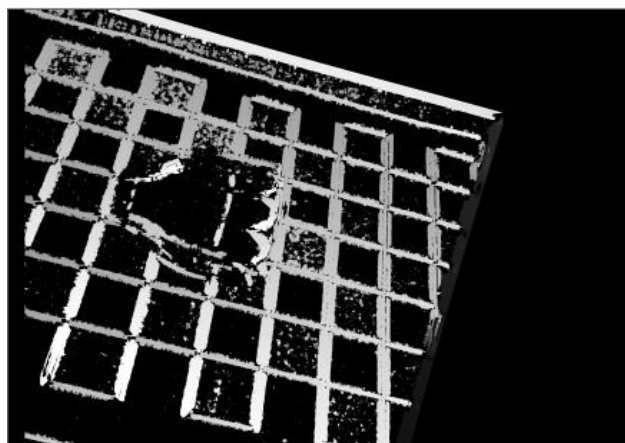


Figura 19. Ejemplo de mapa de disparidad.

Para realizar el procesamiento del mapa de disparidad y obtener una alineación, se buscan puntos de correspondencia en regiones lejanas de ambas imágenes. En dichas regiones la disparidad es cero, por lo tanto las imágenes deben tener iguales coordenadas para dichos puntos, una vez

alineadas las imágenes se podrá definir por el usuario, el valor SAD, que representa el tamaño de búsqueda de puntos de control entre dos imágenes.

El objetivo del algoritmo seleccionado es obtener la correspondencia entre dos puntos del par estéreo y, a partir de ahí, calcular su disparidad que depende en forma inversa de la distancia del sistema de cámaras al objeto. Dada una imagen I_l de dimensiones $N \times M$ y una ventana V_l de dimensiones $n \times m$ en torno a un píxel $l_i(x,y)$ de la imagen izquierda, se busca encontrar una ventana V_d sobre la imagen derecha I_d que sea similar a la anterior. Al encontrar la ventana de mayor similitud se dice que el píxel central $l_i(x,y)$ se corresponde con el píxel central $d(x',y)$ de la ventana derecha. Como las imágenes están rectificadas y se utiliza la restricción epipolar, la búsqueda se realiza sólo sobre la misma línea en la I_d desde la posición (x,y) hasta la posición $(x-d_{max},y)$, donde d_{max} , representa la disparidad máxima. La disparidad se calcula como: $d(x,y)=x-x'$. Si se calcula $d(x,y)$ para cada píxel se obtiene la imagen que representa el mapa de disparidad. Este mapa de disparidad tiene dimensiones $(N-n+1) \times (M-m+1-d_{max})$.

2.8 Obtención de nube de puntos

Para la obtención de la nube de puntos es importante definir los valores adecuados ya que se podrán encontrar con dos problemas típicos:

- ✓ Que apenas se encuentren puntos coincidentes: por ejemplo si se introduce un valor SAD o una disparidad muy bajos, la búsqueda se hará en una zona muy pequeña y puede que las parejas de píxeles no se encuentren en ella.
- ✓ Que encuentre muchos puntos erróneos: si por el contrario se introducen valores excesivos, las zonas de búsqueda serán grandes y es muy posible que el programa empareje píxeles que no tienen nada que ver unos con otros o que se encuentren muy alejados entre sí en la escena.

Para guardar la nube de puntos se tienen varias opciones, las que más interesan para la solución son formato OBJ y formato PLY, estos dos son igual de válidos, ya que todos son archivos de texto con las coordenadas de la nube de puntos, salvo que OBJ no guarda la información de color; dependiendo del proyecto puede que interese tener esa información o no.

2.9 Recursos utilizados en el proceso de desarrollo

Para el entorno de desarrollo de la aplicación se utilizará Visual Studio 2008, C# como lenguaje de programación, la biblioteca OpenCV de Intel en el área procesamiento de imágenes, la metodología de desarrollo RUP y el Visual Paradigm como herramienta CASE.

C# como un lenguaje de programación orientado a objetos que fue creado y estandarizado por Microsoft de manera independiente para desarrollar aplicaciones sobre la plataforma .NET. Su sintaxis se deriva del C/C++ e implementa el modelo de objetos de .NET. Logra capturar los mejores rasgos de los lenguajes de programación más populares que le precedieron, combinándolo en uno. Se valora el uso de este lenguaje debido a su gran flexibilidad, sencillez y seguridad. Además permite trabajar tanto a bajo nivel con el uso de punteros como a un alto nivel, de manera que sea más descriptivo y cercano al lenguaje natural, permite acceder a las características avanzadas de la plataforma sobre la que se trabaja, crear código muy eficiente en aquellos puntos de la aplicación que son críticos y acceder a las interfaces de programación de aplicaciones (API) existentes. Se utilizará **Microsoft Visual Studio.NET 2008**, como entorno integrado de desarrollo (IDE), ya que la misma cuenta con pequeñas herramientas auxiliares de completamiento de códigos como el Visual Assist lo que ayuda mucho a la codificación de la aplicación.

Para la implementación de la aplicación se utilizará la biblioteca **EmguCV**, el cual es un wrapper a la biblioteca **OpenCV** de Intel en el área procesamiento de imágenes. Esta plataforma permite que las funciones de OpenCV puedan ser llamadas desde .NET y sean compatibles con C#. El contenedor se puede compilar en mono y ejecutar en Linux / Mac OS X. Por lo tanto la solución se centra en las funcionalidades de OpenCV como herramienta de desarrollo utilizada, que permite que sea usada libremente para propósitos comerciales y de investigación con las condiciones en ella expresadas. Contiene más de 500 funciones que abarcan una gran gama de áreas en el proceso de visión, como reconocimiento de objetos (reconocimiento facial), calibración de cámaras, visión estéreo y visión robótica. Por lo anteriormente expuesto, el proyecto pretende proporcionar un entorno de desarrollo fácil de utilizar y altamente eficiente. Esto se ha logrado, realizando su programación en código C# optimizado, aprovechando además las capacidades que proveen los procesadores multinúcleo. El proyecto pretende proporcionar un entorno de desarrollo fácil de utilizar y altamente eficiente.

Se utilizará como metodología de desarrollo, **RUP** El Proceso Unificado Racional (Rational Unified Process en inglés), el cual es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado **UML**, constituye unas de las metodologías más utilizadas para el análisis, implementación y documentación de sistemas orientados a objetos. Centrado en la arquitectura y dirigido por casos de uso, utiliza arquitectura basada en componentes y usa modelado de software visual **UML**. Este lenguaje reúne las mejores prácticas de ingeniería de software y que han sido probadas con éxito en el modelado de sistemas grandes y complejos. La representación en UML de un software está formada por las 4+1 vistas de la arquitectura (Vista de casos de uso, Vista lógica, Vista de procesos, Vista de implementación, Vista de despliegue), relacionadas entre sí.

Se utilizará también **Visual Paradigm (VP)** como herramienta CASE, escogida para el desarrollo de esta arquitectura de software para el proyecto. Se seleccionó esta herramienta debido a que posee licencia gratuita y comercial, es un producto de mucha calidad, configurable para el uso en varios idiomas, multilinguaje, muy fácil de instalar y actualizar además de brindar compatibilidad entre ediciones. Esta herramienta soporta modelado **UML** y provee el modelado de procesos de negocios. Visual Paradigm UML responde rápidamente y con poca memoria, permitiendo manejar grandes y complicadas estructuras de un proyecto de forma eficiente y que solo requiera de una configuración de escritorio. Ayuda a los equipos de desarrollo de software a desplegar el proceso de desarrollo de los mismos, logrando maximizar y acelerar tanto las contribuciones individuales como las de equipo, facilita la diagramación visual y el diseño de sus proyectos.

CAPÍTULO 3 Ingeniería del sistema

Introducción

En este capítulo se hará una descripción a nivel conceptual de la solución propuesta. Se definen las condiciones a tener en cuenta para el correcto funcionamiento del sistema, relacionadas como reglas del negocio. Se tratarán los conceptos más importantes del área de interés mediante un modelo de dominio. Se describirán las capacidades que deberá cumplir el sistema a partir de los requisitos funcionales, así como las características del mismo expresadas por los requisitos no funcionales, se definirán los casos de uso del sistema y actores que interactúan con el mismo.

3.1 Características del sistema

En los siguientes epígrafes se enuncia el patrón de arquitectura utilizado, así como las características, capacidades del sistema y el diagrama de casos de uso del sistema.

3.1.1 Patrones de arquitectura

Los patrones arquitectónicos, o patrones de arquitectura, son patrones de software que ofrecen soluciones a problemas de arquitectura de software en ingeniería de software. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones.

Arquitectura de n-capas

Con el estudio realizado, se llegó a la conclusión de que el estilo arquitectónico que más se adapta a la solución de arquitectura es la Arquitectura en Capas, esto ayuda a reducir las dependencias entre componentes de diferentes niveles. La arquitectura propuesta cuenta con dos capas, la capa de Presentación y la de Lógica del Negocio o de Dominio, ya que la utilización de dichas capas proporciona al sistema niveles de abstracción que resuelven muchos problemas de desarrollo, proporcionando reutilización, optimización y refinamiento.

- ✓ **Capa de Presentación:** Esta capa se encarga de gestionar todos aquellos aspectos relacionados con la lógica de presentación de la aplicación. Se comunica con la capa de Negocio a través de la clase FormMain.
- ✓ **Capa de Lógica del Negocio o Dominio:** Contiene las funcionalidades que implementa la aplicación, contiene mecanismos que procesan la información dada por el usuario además de realizar validaciones.

3.1.2 Reglas del negocio

Con el propósito de definir las condiciones que deben satisfacerse para el correcto funcionamiento del sistema de adquisición de modelos geométricos a partir de objetos del mundo real se establecen como reglas del negocio las siguientes:

- ✓ El diseño del entorno debe ser cuadrado o rectangular.
- ✓ Los archivos que representan un proyecto de adquisición serán salvados con extensión .PCP.
- ✓ Los archivos de calibración a cargar deben ser sólo de extensión .txt. Las fotos generadas luego del calculo de distorsión radial serán guardados con el nombre “sindist_” + nombre de la imagen.
- ✓ Las fotos generadas luego de la rectificación será guardada con el nombre “rec_” + nombre dela imagen.
- ✓ La nube de puntos 3D deben ser salvados con la extensión .txt o .obj para su posterior visualización con el nombre “nube3D_” + # de la nube salvada, comenzando por el 000 (Ejemplo: nube3D_000.txt).

3.1.3 Modelo de domino o de negocio

A partir del entendimiento del problema se obtiene el modelo de dominio representado en la Figura 20, que no es más que un acercamiento a la solución propuesta, donde se modelan los principales conceptos así como sus relaciones.

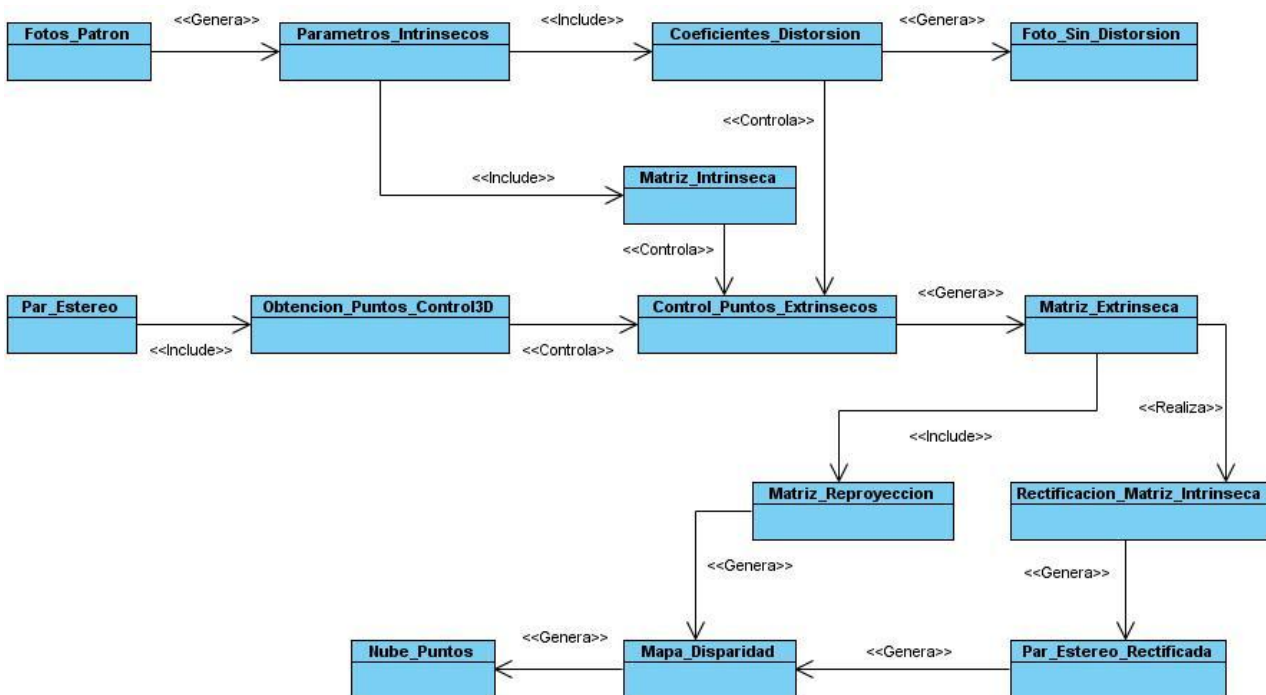


Figura 20. Modelo del dominio.

3.1.4 Glosario de términos

Obtención puntos de control: Empareja en las diferentes imágenes aquellos puntos 2D procedentes de un punto 3D común.

Par estéreo rectificado: Transformar mediante las imágenes obtenidas, conociendo los parámetros tanto intrínsecos como extrínsecos de las cámaras, de manera que las líneas epipolares de ambas imágenes queden alineados horizontalmente.

Parámetros intrínsecos y extrínsecos: Las técnicas de calibración intrínsecas se centran en la obtención de los parámetros físicos y ópticos de la cámara. Por otro lado, las técnicas de calibración extrínsecas miden la posición y la orientación de la cámara con respecto a la escena.

Mapa de disparidad: Las distancias se obtienen a partir de la medición de la disparidad de dos elementos semejantes en ambas imágenes. Las disparidades de todos los puntos de una imagen conforman el denominado mapa de disparidad.

Nube de puntos 3D: Conjunto de puntos tridimensionales obtenidos a partir del proceso de adquisición tridimensional.

3.1.5 Captura de requisitos funcionales

Los requisitos constituyen capacidades o condiciones que el sistema debe cumplir. Estos facilitan el entendimiento entre usuarios y desarrolladores del sistema a elaborar. A continuación se exponen los requisitos funcionales por los que se regirá el sistema y los no funcionales que exponen las características de la aplicación.

Requisitos funcionales

RF1. Gestionar proyecto.

RF1.1. Cargar un proyecto con formato .PCP.

RF1.2. Guardar un proyecto con formato .PCP.

RF1.3. Modificar un proyecto formato .PCP.

RF2.1. Cargar fotos.

RF2.1. Seleccionar directorio donde se encuentran las fotos.

RF2.2. Cargar fotos de calibración.

RF2.3. Cargar fotos estéreo del modelo a restituir.

RF3. Generar parámetros de calibración intrínsecos.

RF3.1. Realizar proceso de detención de esquinas.

RF3.2. Iniciar proceso de calibración.

RF3.3. Seleccionar el directorio para guardar datos de calibración intrínsecos.

RF3.4. Guardar archivos de calibración intrínsecos.

RF4. Generar parámetros de calibración extrínsecos.

RF4.1. Realizar proceso de correspondencia por coordenadas 3D.

RF4.2. Iniciar proceso de calibración por correspondencia 3D.

RF4.3. Seleccionar el directorio para guardar datos de calibración extrínsecos.

RF4.4. Guardar archivos de calibración extrínsecos.

RF5. Cargar parámetros de calibración.

RF5.1. Seleccionar directorio donde se encuentran los archivos de calibración.

RF5.2. Cargar archivos de calibración con formato TXT.

RF6. Obtener puntos de control del par estéreo del modelo.

R6.1. Seleccionar al menos 7 puntos de control.

R6.2. Guardar matriz de proyección generada.

RF7. Generar imágenes sin distorsión radial.

RF7.1. Seleccionar directorio donde desea guardar las nuevas imágenes generadas sin distorsión.

RF7.2. Guardar fotos sin distorsión.

RF8. Generar imágenes rectificadas.

RF8.1. Seleccionar directorio donde desea guardar las nuevas imágenes estéreos rectificadas.

RF8.2. Guardar fotos rectificadas.

RF9. Generar nube de puntos.

RF9.1. Generar mapa de disparidad.

RF9.2. Seleccionar formato de archivo que desea guardar la nube de puntos, OBJ o TXT.

RF9.3. Seleccionar directorio donde desea guardar la nube de puntos.

RF9.4. Guardar nube de puntos.

Requisitos no funcionales

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Entre los requisitos no funcionales que debe tener el sistema a partir de la investigación realizada y modelación se encuentran los siguientes:

Usabilidad

- ✓ La aplicación debe ser de fácil comprensión, navegación, configuración y utilización tanto para usuarios con un nivel alto de experiencia, como de niveles inferiores en el campo de la informática.
- ✓ La información debe ser mostrada de forma lógica y organizada.

Apariencia o interfaz externa

- ✓ El sistema debe brindar una interfaz externa de diseño sencillo y accesible a cualquier tipo de usuario, independientemente de su nivel de conocimiento de la aplicación.
- ✓ El usuario debe tener acceso por varias vías a una determinada funcionalidad.
- ✓ La aplicación debe utilizar como idioma principal el español, excepto aquellas palabras técnicas que no puedan ser traducidas.
- ✓ Utilizar botones que expresen su función, ya sea a través del ícono o el texto que lo identifica.

Soporte

- ✓ Se brindará servicios de instalación y configuración.
- ✓ La aplicación debe estar documentada y proveer el código fuente previendo futuras modificaciones en el mismo para potenciar su alcance o eficiencia.

Rendimiento

- ✓ El sistema debe tener un tiempo de respuesta rápido, de acuerdo con la funcionalidad que se esté usando en cualquier momento.

Requerimientos de Hardware

- ✓ Se requiere una PC Pentium V o superior, con procesamiento mínimo de 2Ghz.
- ✓ Se requiere 1GB de RAM o superior.
- ✓ Se requiere 40GB de disco duro o superior.

Portabilidad

- ✓ El sistema debe ser independiente de plataforma siendo compatible con cualquier sistema operativo.

3.1.6 Modelo de casos de uso del sistema

En esta sección se reconocen los actores del sistema a desarrollar, y se definen los casos de uso del sistema organizados como se observa en la Figura 21. Además, se seleccionan los casos de uso correspondientes al primer ciclo de desarrollo para hacerles sus especificaciones textuales en formato expandido.

Actores del sistema

Los actores de un sistema son agentes externos, roles que las personas (usuarios) o dispositivos juegan cuando interactúan con el software. En este caso particular quien hará uso del sistema

serán los diseñadores de la Línea de Imágenes y Videos basados en Rendering, que como actor del sistema será llamado diseñador.

Actores	Descripción
Diseñador	Es la persona encargada de realizar las operaciones necesarias para adquirir la nube de puntos tridimensionales.

Tabla 2. Actor del sistema.

Diagrama de casos de uso del sistema

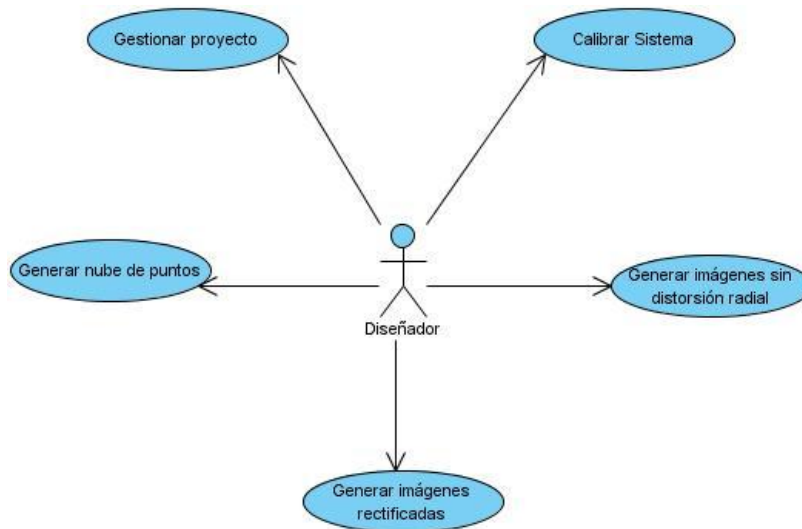


Figura 21. Diagrama de Caso de Uso del Sistema.

Descripción de casos de uso del sistema

Cada caso de uso tiene una descripción de las funcionalidades que ejecutará el sistema propuesto como respuesta a las acciones del usuario. Las tablas presentadas a continuación argumentan los flujos operacionales de cada caso de uso.

Caso de Uso:	Gestionar Proyecto
Actores:	Diseñador
Propósito:	Permite al actor poder crear un nuevo proyecto, así como cargar un proyecto existente y modificarlo.
Resumen:	El caso de uso de inicia cuando el usuario selecciona la opción de cargar, crear nuevo o modificar proyecto. El sistema procede a cargar, crear o modificar los ficheros especificados.
Precondiciones:	

Referencia:	RF1. RF1.1. RF1.2. RF1.3.
Prioridad:	Crítico
Poscondición:	
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. Necesita crear, cargar o modificar un proyecto. <ol style="list-style-type: none"> a. Si desea crear un nuevo proyecto, ir a la sección “Crear Nuevo proyecto”. b. Si desea cargar un proyecto existente, ir a la sección “Cargar proyecto”. c. Si desea modificar un proyecto existente, ir a la sección “Modificar proyecto”. 	
Sección “Crear nuevo proyecto”	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. Entra el nombre del nuevo proyecto. 	<ol style="list-style-type: none"> 1.1 Verifica que el nombre del proyecto no exista. 1.2 Se almacena el proyecto. 1.3 Muestra el nombre del proyecto cargado.
Sección “Cargar proyecto”	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 2. Selecciona el nombre del proyecto a cargar. 	<ol style="list-style-type: none"> 2.1 Carga el archivo seleccionado. 2.2 Muestra el nombre del proyecto cargado.
Sección “Modificar proyecto”	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. Selecciona el nombre del proyecto a 	<ol style="list-style-type: none"> 1.1 Carga el archivo seleccionado.

cargar.	1.2 Muestra el nombre del proyecto cargado.
2. Realiza cualquier funcionalidad.	2.1 Modifica los valores cargados.
3. Selecciona opción guardar.	3.1 Guarda los datos modificados.

Tabla 3. CU Gestionar proyecto.

Caso de Uso:	Calibrar Sistema	
Actores:	Diseñador	
Propósito:	Permite al actor obtener los parámetros intrínsecos y extrínsecos de la cámara, obteniendo parámetros como distancia focal, distancia al centro de la imagen, traslación y rotación, parámetros de vital importancia en el proceso de adquisición tridimensional.	
Resumen:	El caso de uso inicia cuando el actor carga las imágenes del patrón para realizar la detección de esquinas y obtener la calibración intrínseca de la cámara, luego carga las imágenes del objeto a restituir, obtiene los puntos de control para obtener los parámetros de calibración extrínsecos de las cámaras para el trabajo de mediciones de fotografías, el sistema guarda estos parámetros para su posterior utilización.	
Precondiciones:	Obtener los pares estéreos de calibración y el par estéreo del objeto a restituir.	
Referencia:	RF2. RF2.1. RF2.2. RF2.3. RF3. RF3.1. RF3.2. RF3.3. RF3.4. RF4. RF4.1. RF4.2. RF4.3. RF4.4 RF5. RF5.1. RF5.2. RF6. RF6.1. RF6.2.	
Prioridad:	Crítico	
Poscondición:	Obtener los parámetros intrínsecos y extrínsecos de la cámara.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
<ol style="list-style-type: none"> 1. Necesita obtener parámetros de calibración del sistema. <ol style="list-style-type: none"> a. Si desea generar los parámetros de calibración intrínsecos, ir a la sección “Generar parámetros de calibración intrínsecos”. b. Si desea generar los parámetros de calibración extrínsecos, ir a la sección “Generar parámetros de calibración 		

extrínsecos”.	
c. Si desea cargar los parámetros de calibración, ir a la sección “Cargar parámetros de calibración”.	
Sección “Generar parámetros de calibración intrínsecos”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción cargar imágenes de patrón.	1.1 Carga las imágenes de calibración.
2. Selecciona la acción de detección de esquinas de las imágenes cargadas.	2.1 Realiza la detención de esquinas.
3. Selecciona opción de calibración.	3.1 Calibra la cámara obteniendo sus parámetros intrínsecos.
4. Selecciona guardar parámetros de calibración intrínsecos generados.	4.1 Muestra directorio donde desea guardar los datos de calibración intrínseca.
5. Selecciona directorio y acepta para guarda los parámetros en archivos de texto plano.	5.1 Guarda los parámetros y almacena en el sistema para su posterior utilización.
Sección “Generar parámetros de calibración extrínsecos”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción cargar imágenes del objeto a restituir.	1.1 Carga las imágenes del objeto.
2. Selecciona opción establecer correspondencia 3D.	2.1 Muestra información detallada de las imágenes de par estéreo del modelo a restituir.
3. Selecciona los puntos que corresponde a un mismo punto 3D común.	3.1 Si la cantidad de puntos de correspondencia es mayor que 7, guarda dicha información de correspondencia estéreo, sino muestra un mensaje “Faltan puntos de correspondencia”.
4. Selecciona la opción salvar datos generados.	4.1 Guarda los datos generados para su posterior utilización.
5. Selecciona la opción calibrar extrínseco mediante calibración de coordinación 3D.	5.1 Calibra la cámara obteniendo sus parámetros extrínsecos.

6. Selecciona guardar parámetros de calibración extrínsecos generados.	6.1 Muestra directorio donde desea guardar los datos de calibración extrínseca.
7. Selecciona directorio y acepta para guarda los parámetros en archivos de texto plano.	7.1 Guarda los parámetros y almacena en el sistema para su posterior utilización.
Sección "Cargar parámetros de calibración"	
Acción del Actor	Respuesta del Sistema
1. Selecciona opción cargar parámetros de calibración intrínsecos.	1.1 Muestra directorio.
2. Selecciona los parámetros de calibración que desea cargar.	2.1 Interpreta estos parámetros y los almacena en su sistema para su posterior utilización.
3. Selecciona opción cargar parámetros de calibración extrínsecos.	3.1 Muestra directorio.
4. Selecciona los parámetros de calibración que desea cargar.	4.1 Interpreta estos parámetros y los almacena en su sistema para su utilización.

Tabla 4. CU Calibrar Sistema.

Caso de Uso:	Generar imágenes sin distorsión radial.
Actores:	Diseñador
Propósito:	Permite al actor guardar el par de imágenes estéreo del modelo a restituir sin distorsión radial.
Resumen:	El caso de uso inicia cuando el actor selecciona la opción de generar imágenes corregidas mediante la distorsión radial, el sistema es capaz de guardar dichas imágenes en el directorio seleccionado.
Precondiciones:	Haber obtenido los parámetros intrínsecos de la cámara.
Referencia:	RF7. RF7.1. RF7.2.
Prioridad:	Crítico
Poscondición:	Imágenes sin distorsión radial.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción generar	1.1 Muestra el directorio donde están las imágenes

fotos sin distorsión radial.	con distorsión radial.
2. Selecciona la imagen derecha a corregir.	2.1 Corrige la imagen izquierda y muestra un mensaje de “corrección de distorsión radial con éxito”. 2.2 Guarda la imagen generada sin distorsión radial.
3. Selecciona la imagen izquierda a rectificar.	3.2 Corrige la imagen derecha y muestra un mensaje de “corrección de distorsión radial con éxito”. 3.3 Guarda la imagen generada sin distorsión radial.

Tabla 5. CU Generar imágenes sin distorsión radial.

Caso de Uso:	Generar imágenes rectificadas.	
Actores:	Diseñador	
Propósito:	Permite al actor guardar el par de imágenes estéreo del modelo a restituir rectificadas según su traslación y rotación.	
Resumen:	El caso de uso inicia cuando el actor selecciona la opción de rectificación de imágenes, el sistema es capaz de guardar dichas imágenes en el directorio seleccionado.	
Precondiciones:	Haber calibrado el sistema, o sea ya el sistema esta previamente calibrado y adquirido los parámetros intrínsecos y extrínsecos de la cámara.	
Referencia:	RF8. RF8.1. RF8.2.	
Prioridad:	Crítico	
Poscondición:	Imágenes rectificadas, o sea las líneas epipolares alineadas horizontalmente.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. Selecciona la opción generar fotos rectificadas.	1.1 Muestra el directorio donde están las imágenes que aun no han sido rectificadas.	
2. Selecciona la imagen izquierda a rectificar.	2.1 Corrige la imagen izquierda y muestra un mensaje de “rectificación realizada con éxito”. 2.2 Guarda la imagen generada rectificada. 2.3 Muestra el directorio donde están las imágenes	

	que aun no han sido rectificadas.
3. Selecciona la imagen derecha a rectificar.	<p>3.1 Corrige la imagen derecha y muestra un mensaje de “rectificación realizada con éxito”.</p> <p>3.2 Guarda la imagen generada rectificada.</p>

Tabla 6. CU Generar imágenes rectificadas.

Caso de Uso:	Generar nube de puntos 3D.	
Actores:	Diseñador	
Propósito:	Permite al actor seleccionar las imágenes previamente rectificadas para obtener su mapa de disparidad, a partir del mapa de disparidad se obtienen los puntos en profundidad.	
Resumen:	El caso de uso inicia cuando el actor selecciona el par estéreo del modelo previamente rectificado y genera el mapa de disparidad, el sistema es capaz de generar dicho mapa, realizar cálculos de profundidad y ofrecer dichas coordenadas traducidas como nube de puntos 3D donde puede se guarda en un directorio seleccionado.	
Precondiciones:	Imágenes rectificadas.	
Referencia:	RF9. RF9.1. RF9.2. RF9.2. RF9.3. RF9.4.	
Prioridad:	Crítico	
Poscondición:	Mapa de disparidad y nube de puntos del objeto a restituir.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. Selecciona opción generar nube de puntos.	1.1 Muestra la ventana del procedimiento para la adquisición de la nube de puntos.	
2. Carga la imágenes, izquierda y derecha del par estéreo del modelo previamente rectificadas	2.1 Muestra directorio donde están dichas imágenes.	
3. Genera nube de puntos.	3.1 Muestra mapa de disparidad y muestra en que formato desea guardar la nube de puntos.	
4. Selecciona el formato que desea guardar la nube de puntos	4.1 Guarda las coordenadas tridimensionales del objeto y muestra la cantidad de vértices guardados.	

Tabla 7. CU Generar nube de puntos 3D.

3.2 Diseño del sistema

El diseño es el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción, esto contribuye a una arquitectura estable y sólida. En el diseño se modela el sistema y se encuentra su forma para que soporte todos los requisitos, incluyendo los no funcionales y las restricciones que se le suponen. Una entrada esencial en el diseño es el resultado del análisis, que proporciona una comprensión detallada de los requisitos.

3.2.1 Patrones de diseño

Un patrón de diseño nombra, abstrae e identifica los aspectos claves de un diseño estructurado común, que lo hace útil para la creación de diseños orientados a objetos reutilizables. En este sistema se utilizan los patrones singleton, creador, bajo acoplamiento y alta cohesión.

Singleton: Se empleará este patrón de diseño GOF ya que garantiza que exista una instancia única para la clase RectificacionEstereo y proporciona un punto de acceso global a ella. A partir de esta instancia se podrán controlar más fácilmente el acceso a esta clase que tiene responsabilidades bien definidas.

Creador: El patrón creador es un patrón GRASP que guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que se conecte con el objeto producido en cualquier evento. Al escogerlo como creador, se da soporte al bajo acoplamiento. En este caso la clase Proyecto es la responsable de la creación de los parámetros intrínsecos y extrínsecos de la cámara.

En el diseño de clases propuesto se hace uso también de los patrones bajo acoplamiento y alta cohesión ya que las clases son independientes y con responsabilidades bien definidas y los objetos colaboran para realizar las tareas más complejas.

Bajo Acoplamiento: Este patrón estimula asignar una responsabilidad de modo que su colocación no incremente el acoplamiento produciendo resultados negativos propios de un alto acoplamiento. El bajo acoplamiento soporta el diseño de clases más independientes, que reducen el impacto de los cambios.

Alta Cohesión: Se da una alta cohesión cuando los elementos de un componente, una clase por ejemplo, colaboran para producir algún comportamiento bien definido. Una clase de alta cohesión posee un número relativamente pequeño de responsabilidades, con una importante funcionalidad relacionada y poco trabajo por hacer.

3.2.2 Diagramas de clases del diseño del sistema

El diagrama de clases del diseño se basa en el diagrama anterior y se modifica para satisfacer los detalles de las implementaciones, este diagrama muestra como queda compuesta la estructura de clases que se emplea en la implementación del sistema. En la Figura 22 se aprecia la estructura de clases para el sistema de adquisición tridimensional.

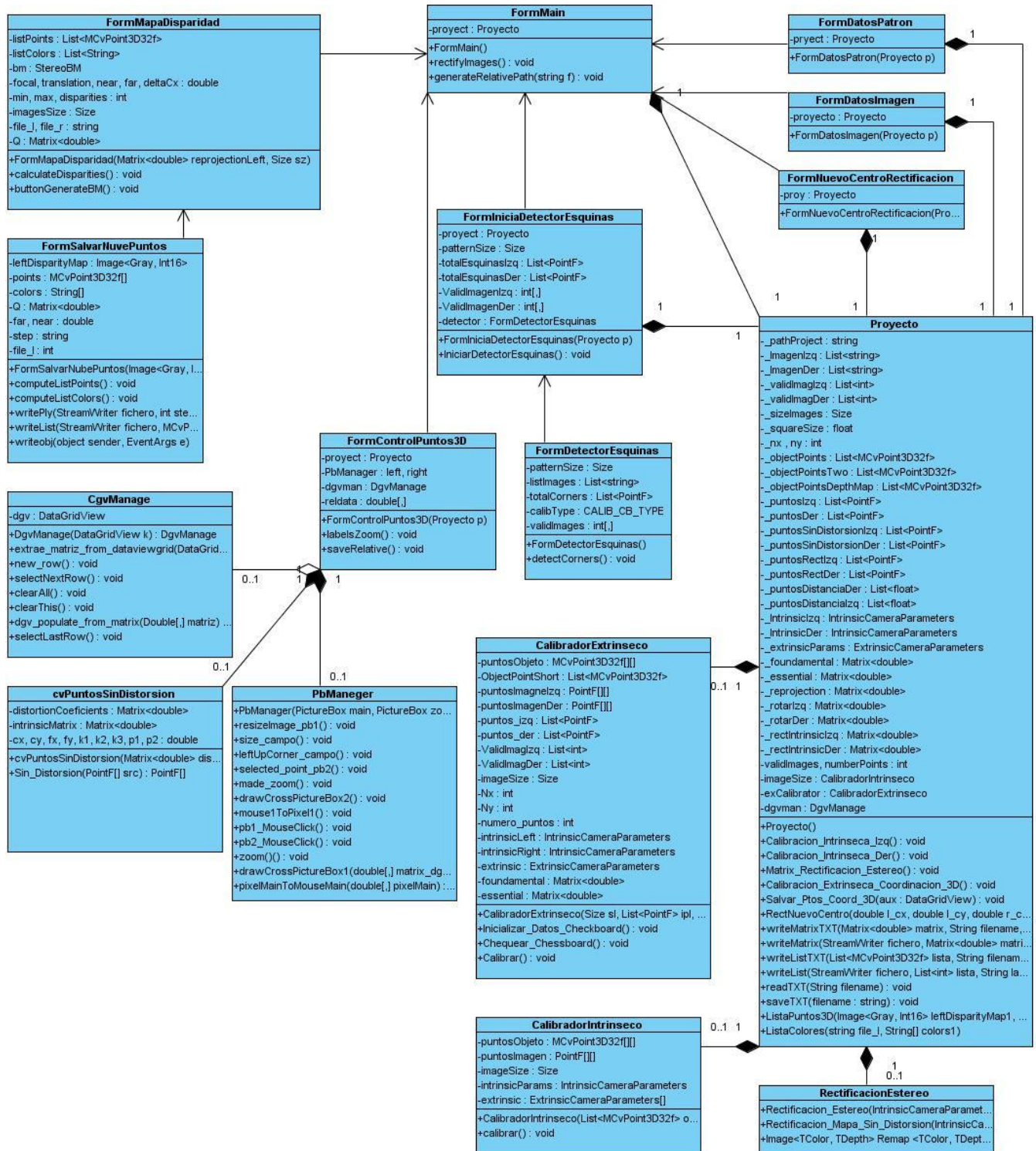


Figura 22. Diagrama de Clases del Diseño.

3.2.3 Descripción de las clases de diseño

Nombre: Proyecto	
Tipo de Clase: Controladora	
Atributo	Tipo
ImagenIzq	List<string>
ImagenDer	List<string>
validImagIzq	List<int>
validImagDer	List<int>
sizeImages	System.Drawing.Size
squareSize	float
nx , ny	int
objectPoints	List<MCvPoint3D32f>
objectPointsDepthMap	List<MCvPoint3D32f>
puntosIzq , puntosDer	List<PointF>
puntosSinDistorsionIzq	List<PointF>
puntosSinDistorsionDer	List<PointF>
puntosRectIzq , puntosRectDer	List<PointF>
IntrinsicIzq , IntrinsicDer	IntrinsicCameraParameters
extrinsicParams	ExtrinsicCameraParameters
fundamental	Matrix<double>
esencial	Matrix<double>
reprojection	Matrix<double>
rotarIzq, rotarDer	Matrix<double>
rectIntrinsicIzq, rectIntrinsicDer	Matrix<double>
numberPoints	int
imageSize	Size
intrinsicCal	CalibradorIntrinseco
exCalibrator	CalibradorExtrinseco
dgvman	DgvManage
rec estereo	RectificacionEstereo
Para cada responsabilidad:	
Nombre:	public Proyecto ()
Descripción:	Constructor.
Nombre:	public void Calibracion_Intrinseca_Izq ()
Descripción:	Realiza la calibración intrínseca de la imagen tomada por la cámara izquierda.
Nombre:	public void Calibracion_Intrinseca_Der ()
Descripción:	Realiza la calibración intrínseca de la imagen tomada por la cámara derecha.
Nombre:	public void Matrix_Rectificacion_Estereo ()
Descripción:	Se obtiene la matriz de rectificación estéreo.
Nombre:	public void Calibracion_Extrinseca_Coordinacion_3D
Descripción:	Obtiene los parámetros de calibración extrínseca según las matrices de rotación y traslación generadas en la coordinación estéreo.
Nombre:	public void Salvar_Ptos_Coord_3D
Descripción:	Salva los puntos de coordinación 3D generados en el Datagridview.
Nombre:	public void RectNuevoCentro(double l_cx, double l_cy, double r_cx, double r_cy, double focal)
Descripción:	Rectifica los atributos que contienen el centro de la imagen y la distancia focal.
Nombre:	public void DefaultIntrinsic(double focal)
Descripción:	Según la distancia focal, calcula los valores intrínsecos por default.
Nombre:	public void ExtrinsicFromRightLeftRotations(double tx)
Descripción:	Calcula la matriz extrínseca a partir de la rotación entre la imagen derecha e izquierda.
Nombre:	public void writeMatrixTXT(Matrix<double> matrix, String filename, String label)

Descripción:	Escribe las matrices de calibración en archivos .txt diferentes.
Nombre:	<code>public void writeListTXT(List<MCvPoint3D32f> lista, String filename, String label)</code>
Descripción:	Escribe la lista de los parámetros a representar de las cámaras.
Nombre:	<code>private void writeList(StreamWriter fichero, List<MCvPoint3D32f> lista, String label)</code>
Descripción:	Escribe lista de puntos obtenidos en el fichero txt.
Nombre:	<code>public void readTXT(String filename)</code>
Descripción:	Leer los datos generados previamente de un archivo txt.
Nombre:	<code>public void saveTXT(String filename)</code>
Descripción:	Salvar los datos generados en un archivo txt.
Nombre:	<code>public void ListaColores(string file_l, String[] colorsl)</code>
Descripción:	Salva el color del píxel del punto 2D correspondiente a un punto 3D común.
Nombre:	<code>public void ListPontos3D(Image<Gray, Int16> leftDisparityMap1, Matrix<double> Q1, MCvPoint3D32f[] points1)</code>
Descripción:	Salva los puntos tridimensionales a partir del mapa de disparidad generado.

Tabla 8. Descripción de la clase Proyecto

Nombre: RectificacionEstereo	
Tipo de Clase: Entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	<code>public static void Rectificacion_Estereo(IntrinsicCameraParameters intrinsicParam1, IntrinsicCameraParameters intrinsicParam2, System.Drawing.Size imageSize, ExtrinsicCameraParameters extrinsicParams, out Matrix<double> R1, out Matrix<double> R2, out Matrix<double> P1, out Matrix<double> P2, out Matrix<double> Q, STEREO RECTIFY TYPE flags)</code>
Descripción:	Constructor.
Nombre:	<code>public static void Rectificacion_Mapa_Sin_Distorsion(IntrinsicCameraParameters intrinsicParam, Matrix<double> R, System.Drawing.Size imageSize, Matrix<double> newCameraMatrix, out Matrix<float> mapx, out Matrix<float> mapy)</code>
Descripción:	Además de la corrección de distorsión de la lente, la función también se puede aplicar arbitrariamente la perspectiva de transformación y, finalmente la escala y puede cambiar la imagen de acuerdo con la cámara nueva matriz.
Nombre:	<code>public static Image<TColor, TDepth> Remap <TColor, TDepth>(Image<TColor, TDepth> sourceImage, Matrix<float> mapx, Matrix<float> mapy)</code>
Descripción:	Rectifica una imagen utilizando su mapa de rectificación.

Tabla 9. Descripción de la clase RectificacionEstereo.

Nombre: CalibradorExtrinseco	
Tipo de Clase: Entidad	
Atributo	Tipo
puntosObjeto	MCvPoint3D32f[] []

objectPointsShort	List<MCvPoint3D32f>
puntosImagenIzq, puntosImagenDer	PointF[][]
puntos_izq, puntos_der	List<PointF>
ValidImagIzq, ValidImagDer	List<int>
imageSize	Size
Nx, Ny, numero puntos	int
intrinsicDer, intrinsicIzq	IntrinsicCameraParameters
extrinsic	ExtrinsicCameraParameters
fundamental	Matrix<double>
esencial	Matrix<double>
Para cada responsabilidad:	
Nombre:	public CalibradorExtrinseco(Size sI, List<PointF> ipl, List<PointF> ipr, List<MCvPoint3D32f> op, IntrinsicCameraParameters ir, intrinsicCameraParameters il)
Descripción:	Constructor.
Nombre:	private void Inicializar_Datos_Checkboard()
Descripción:	Inicializa los datos del patrón.
Nombre:	private void Chequear_Chessboard()
Descripción:	Chequea cada esquina del patrón.
Nombre:	public void Calibrar()
Descripción:	Realiza la calibración extrínseca mediante el método de calibración de Emgucv.

Tabla 10. Descripción de la clase CalibradorExtrinseco.

Nombre: CalibradorIntrinseco	
Tipo de Clase: Entidad	
Atributo	Tipo
puntosObjeto	MCvPoint3D32f[][]
puntosImagen	PointF[][]
imageSize	Size
intrinsicParams	IntrinsicCameraParameters
extrinsic	ExtrinsicCameraParameters[]
Para cada responsabilidad:	
Nombre:	public CalibradorIntrinseco(List<MCvPoint3D32f> object_p, List<PointF> image_p, int nImages, int nPoints, Size image_s, IntrinsicCameraParameters intrinsic_p)
Descripción:	Constructor.
Nombre:	public void calibrar()
Descripción:	Realiza la calibración intrínseca mediante el método de calibración de Emgucv.

Tabla 11. Descripción de la clase CalibradorIntrinseco.

Nombre: cvPuntosSinDistorsion	
Tipo de Clase: Entidad	
Atributo	Tipo
distortionCoeficients	Matrix<double>
intrinsicMatrix	Matrix<double>
cx, cy, fx, fy, k1, k2, k3, p1, p2	double
Para cada responsabilidad:	
Nombre:	public cvPuntosSinDistorsion(Matrix<double> distCoef, Matrix<double> intMatrix)
Descripción:	Constructor.
Nombre:	public PointF[] Sin_Distorsion(PointF[] src)
Descripción:	Calcula la distorsión radial de todos los puntos.

Tabla 12. Descripción de la clase cvPuntosSinDistorsion.

Nombre: DgvManege

Tipo de Clase: Entidad	
Atributo	Tipo
dgv	DataGridView
Para cada responsabilidad:	
Nombre:	<code>public DgvManage (DataGridView k)</code>
Descripción:	Constructor.
Nombre:	<code>public double[,] extrae_matriz_from_dataviewgrid(DataGridView k)</code>
Descripción:	Extraer matriz de datos de un datagridview.
Nombre:	<code>public void new_row()</code>
Descripción:	Agrega los datos seleccionados a la nueva fila del datagridview.
Nombre:	<code>public void selectNextRow()</code>
Descripción:	Busca la fila activa actual.
Nombre:	<code>public void clearAll()</code>
Descripción:	Borra el todo el contenido del datagridview.
Nombre:	<code>public void clearThis()</code>
Descripción:	Borra el contenido de la fila seleccionada del datagridview.
Nombre:	<code>public void dgv_populate_from_matrix(Double[,] matriz)</code>
Descripción:	Llena un datagridview a partir de una matriz double.

Tabla 13. Descripción de la clase DgvManege.

Nombre: PbManeger	
Tipo de Clase: Entidad	
Atributo	Tipo
pb1, pb2	PictureBox
imgpath	string
imgBmp	Bitmap
Campo, pixel1, pixel2, pixelzoom, mouse1, mouse2	double[]
zoomf	Double
Para cada responsabilidad:	
Nombre:	<code>public PbManager (PictureBox main, PictureBox zoom, String img)</code>
Descripción:	Constructor
Nombre:	<code>public void resizeImage_pb1()</code>
Descripción:	Esta función permite re escalar la imagen al valor más cercano de newW o de newH en función de las dimensiones de la imagen.
Nombre:	<code>public void size_campo()</code>
Descripción:	Calcula el tamaño del campo que quiere ver el usuario (ancho y alto) con el valor del factor de zoom elegido.
Nombre:	<code>public void leftUpCorner_campo()</code>
Descripción:	Calcula las coordenadas pixel en la imagen original y1 hace zoom en esa zona. Toma las coordenadas mouse de una selección del usuario en el picture box1 y calcula que el tamaño de campo.
Nombre:	<code>public void selected_point_pb2()</code>
Descripción:	Calcula las coordenadas en píxel en la imagen original del punto pulsado por el usuario en el PB2 (zoom), a partir de las coordenadas del ratón en ese pb2 y de las coordenadas píxel de la esquina superior izquierda de la zona vista en zoom en ese momento.
Nombre:	<code>private void made_zoom()</code>
Descripción:	Hacer zoom toma el píxel superior izquierdo de la zona.
Nombre:	<code>public double[,] pixelMainToMouseMain(double[,] pixelMain)</code>
Descripción:	Obtiene el factor de transformación entre picture box1 e imagen.

Tabla 14. Descripción de la clase PbManeger.

Diagramas de interacción del diseño

Los diagramas de interacción incluyen la interacción de los mensajes entre los objetos que se definen en el modelo conceptual y otras clases de objetos. Un diagrama de secuencia muestra una interacción que está organizada como una secuencia temporal. En particular, muestra los objetos que participan en la interacción mediante sus líneas de vida y mediante los mensajes que intercambian, organizados en forma de una secuencia temporal. (21). A continuación se muestran los diagramas de secuencias correspondientes al sistema para la adquisición de modelos geométricos a partir de un objeto del mundo real.

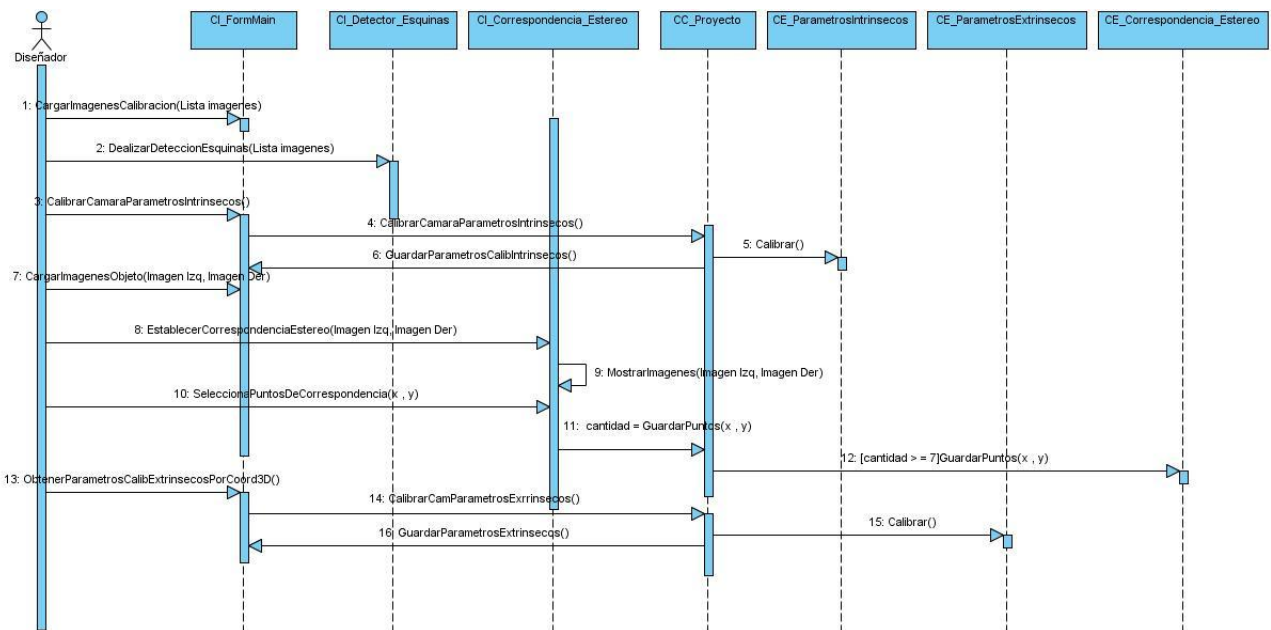


Figura 23. Diagrama de Secuencia: CU Calibrar sistema.

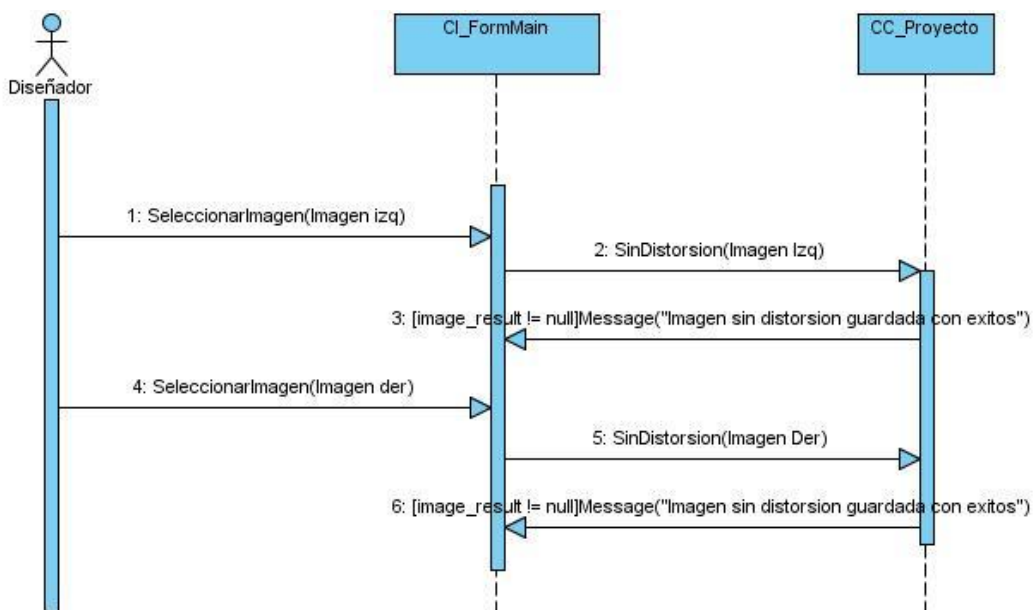


Figura 24. Diagrama de Secuencia: CU Generar imágenes sin distorsión radial.

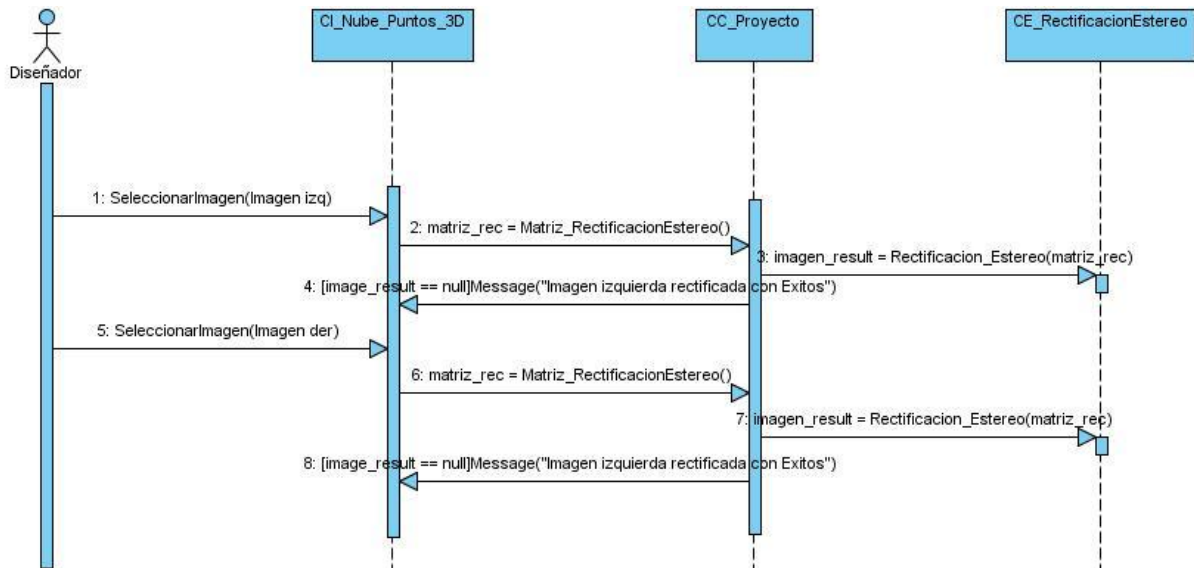


Figura 25. Diagrama de Secuencia: CU Generar imágenes rectificadas.

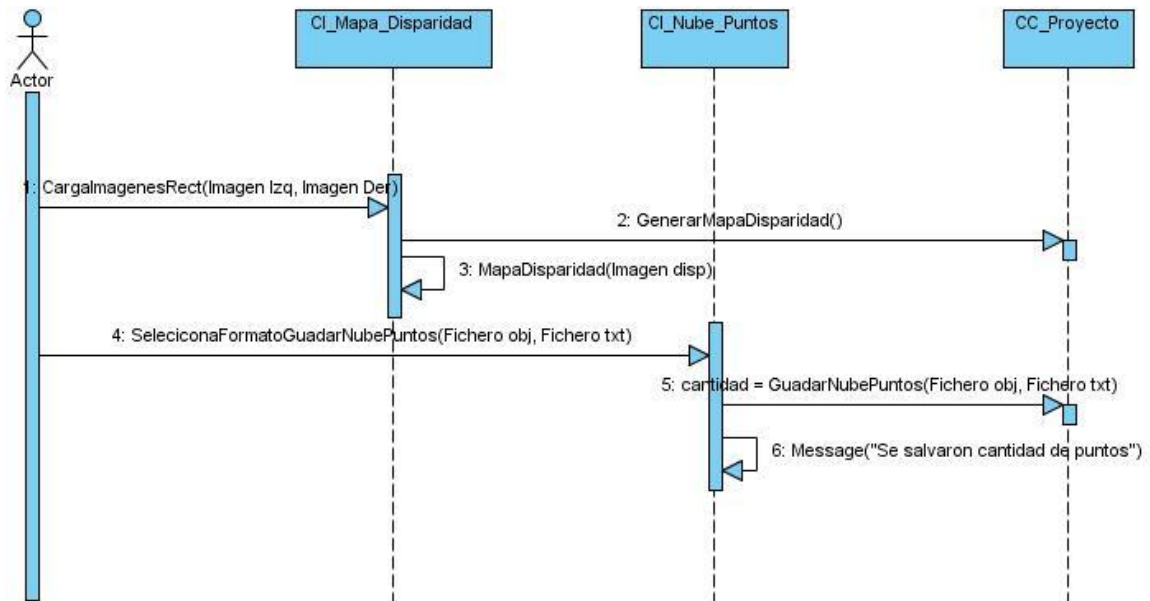


Figura 26. Diagrama de Secuencia: CU Generar nube puntos 3D.

Capítulo 4. Implementación y prueba

Introducción

En este capítulo se abordan los temas de la implementación del prototipo basados en todo el trabajo acumulado a lo largo de los capítulos anteriores, y además se hará un análisis de los resultados del sistema, en cuanto a rendimiento, cumplimiento de los objetivos propuestos, entre otras cosas.

4.1 Implementación

Correspondientes a esta etapa se muestra el diagrama de componentes, así como los estándares y notaciones utilizados en la implementación de la aplicación.

4.1.1 Diagrama de componente

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre sus elementos. En la Figura 27 se muestra el Diagrama de componentes correspondiente al sistema desarrollado.

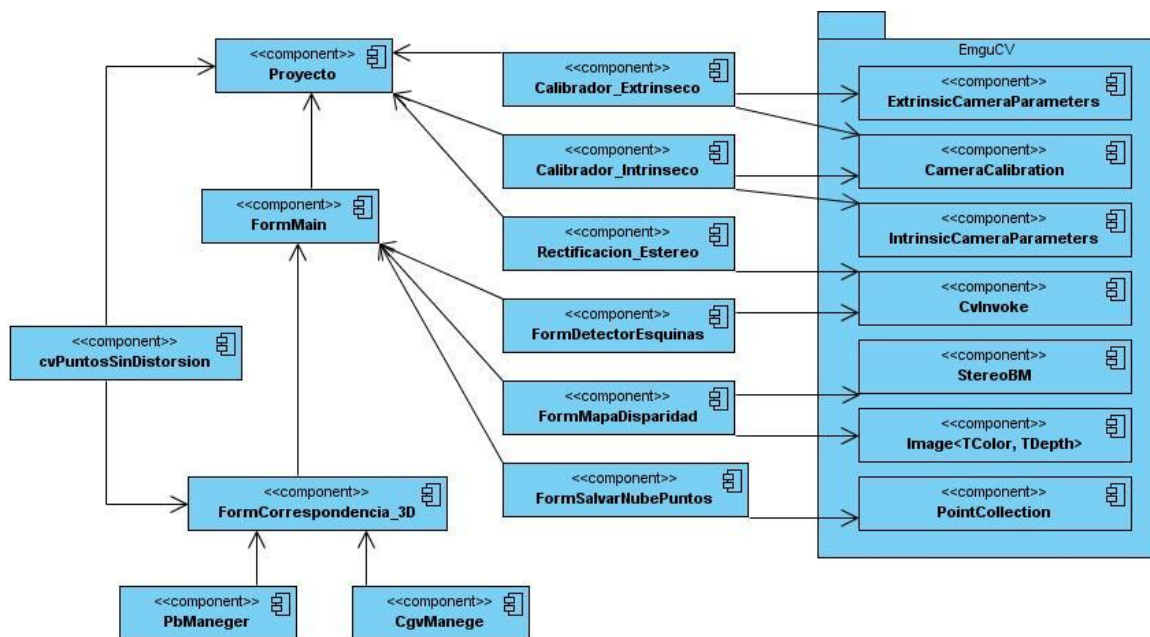


Figura 27. Diagrama de Componente.

4.2 Prueba

Una vez desarrollados los componentes gráficos, básicos y avanzados, es necesario realizarles pruebas, debido a que su funcionamiento correcto no implica que no carezcan de fallas. Es por ello que este epígrafe se centra en las pruebas para validar cada una de las soluciones que se dieron en la implementación.

4.2.1 Fase de prueba

La fase de prueba en el desarrollo de software consistente en probar las aplicaciones construidas. Las pruebas permiten verificar, validar y revelar la calidad de un producto software. La prueba es el proceso de ejecución de un programa con la intención de descubrir un error. Gran porcentaje de los programas que se desarrollan tienen errores, en esta fase de pruebas es donde se encuentran y corrigen generalmente los errores de las fases anteriores. Es por ello que probar es una de las fases más importantes para obtener un producto final con la calidad requerida.

4.2.2 Recursos empleados en la realización de las pruebas

Para realizar las pruebas se emplearon recursos como una computadora con un microprocesador AMD Sempron, con una velocidad de CPU de 2,20 Ghz, que posee una memoria RAM de 1 Gb y un disco duro de 160 Gb. Se utilizó una cámara fotográfica (BENQ DC C1060 10 Mega Píxel) para la adquisición de los pares estéreo. Se empleó además, el sistema operativo Windows, la biblioteca de clases EmguCV para automatizar el proceso y el software Rapidform 2006 Basic que permite visualizar la nube de puntos obtenida, el cual se encuentra completamente gratuito en Internet.

4.3 Pruebas desarrolladas

A continuación se muestran aspectos que representan los parámetros a medir para la realización de las pruebas al software en los que se exponen elementos fundamentales como datos de entrada y valoración de los resultados en cada uno de ellos, como parte del tipo de pruebas utilizado, el cual se centra en las pruebas de caja negra, y entre otros aspectos se probó la respuesta del sistema ante valores extremos e insuficiencia de datos.

1. Adquisición de pares estéreo para la calibración de la cámara.
2. Imágenes con detección de esquinas correctas.
3. Adquisición del par estéreo del objeto a restituir.
4. Correspondencia estéreo.
5. Rectificación de imágenes.

6. Imagen del mapa de profundidades según el tamaño SAD.
7. Datos de la nube de puntos obtenida.
 - ✓ La calidad tomada (SAD 61, SAD 41...): se intentará obtener las superficies con buena calidad (SAD 41 ó 61), pero esta se bajará si no se obtienen suficientes puntos.
NOTA: la característica SAD indica el tamaño de las ventanas en las que el programa busca puntos coincidentes.
 - ✓ Los puntos guardados, 1 de cada 1, 1 de cada 10, 1 de cada 100, en principio se guardarán en estas tres modalidades.
8. Puntos totales de la Nube
9. Imagen con una perspectiva de la nube de punto con mejores resultados obtenidos.

En la Figura 28 se muestran algunas imágenes tomadas para la calibración, las cuales se realizaron en condiciones de luz natural, con veinte tomas de diferentes ángulos para garantizar la efectividad y el menor error posible en la obtención de los datos de calibración.



Figura 28. Fotos para la calibración.

Del total de imágenes obtenidas, se efectuó la detección de esquinas y se lograron definir 19 imágenes correctas para dicho proceso para un 90,4% de efectividad en dicho proceso. Por lo que no se tendrá que realizar este proceso nuevamente siempre que se utilice la misma cámara ya que estos datos de calibración pueden ser guardados y cargados para su posterior utilización.

Las matrices de calibración interna obtenidas por la aplicación fueron comparadas con los resultados ofrecidos utilizando el software de calibración GML Camera Calibration Toolbox v.0.4, donde se mostraron valores semejantes en cuanto distancia focal y centro de la imagen. En la Figura 29 se muestra la detección de esquinas en el patrón para obtención de dichos parámetros producidas por el prototipo desarrollado.

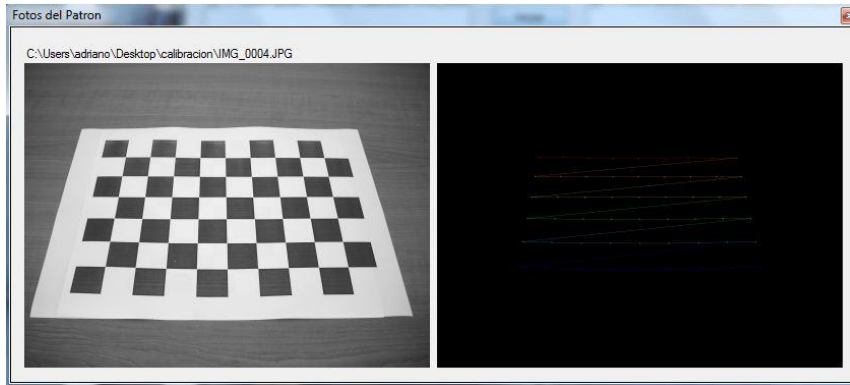


Figura 29. Detención de esquinas.

Se realizaron varias tomas estéreas al objeto a restituir, con el objetivo de tomar las que mejor enfoque y luminosidad tenían. Ejemplo de estas tomas estéreas se muestra en la figura 30.

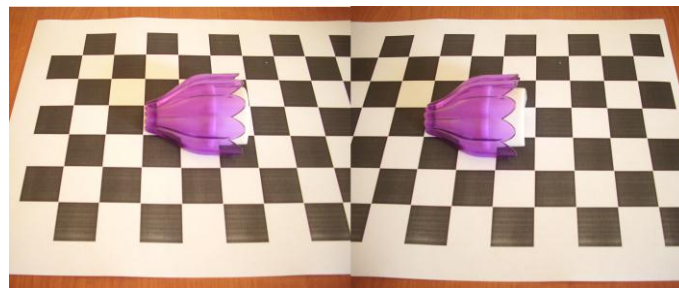


Figura 30. Par estéreo del objeto.

A partir del par estéreo para la adquisición de la nube de puntos se pasa a realizar la correspondencia estéreo para obtener los parámetros de calibración extrínseca. Se obtuvieron 15 puntos de correspondencia estéreo manualmente. En la Figura 31 se muestran imágenes que indican el desarrollo de dicho proceso.

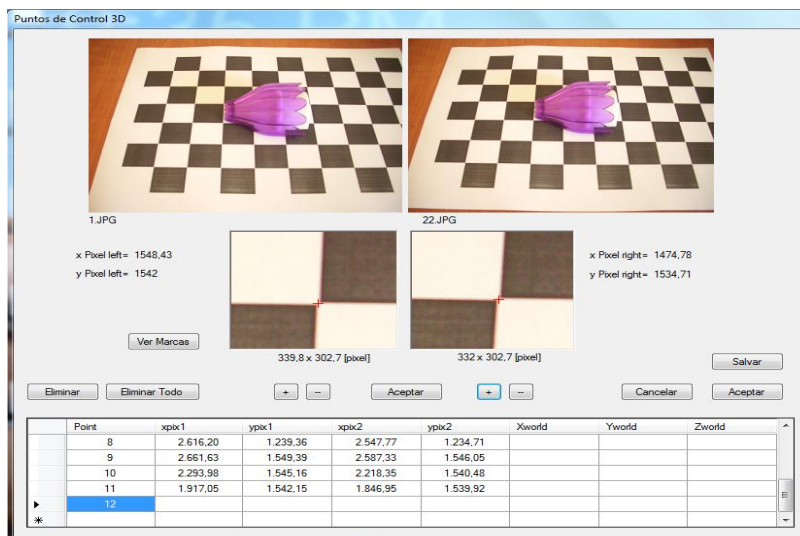


Figura 31. Obtención de puntos de control.

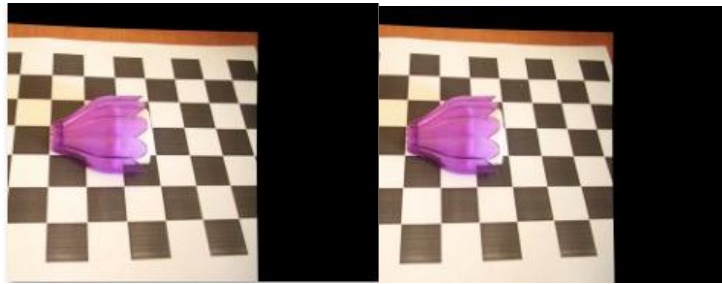


Figura 32. Imágenes estéreas rectificadas.



Figura 33. Mapa de disparidad para SAD = 21.

Caso de Uso	Caso de Prueba	Entrada	Resultado(Puntos)	Condiciones
Obtener nube de puntos	Adquisición geometría 3D	1 punto de cada 100, Tamaño SAD= 21	7810	Generado el mapa de disparidad
		1 punto de cada 10, Tamaño SAD= 21	78093	Generado el mapa de disparidad
		1 punto de cada 1, Tamaño SAD= 21	1883435	Generado el mapa de disparidad
		1 punto de cada 100, Tamaño SAD= 41	9810	Generado el mapa de disparidad
		1 punto de cada 10, Tamaño SAD= 41	80936	Generado el mapa de disparidad
		1 punto de cada 1, Tamaño SAD= 41	345435	Generado el mapa de disparidad
		1 punto de cada 100, Tamaño SAD= 61	5619	Generado el mapa de disparidad
		1 punto de cada 10, Tamaño SAD= 61	90932	Generado el mapa de disparidad
		1 punto de cada 1, Tamaño SAD= 61	683435	Generado el mapa de disparidad

Tabla 15. Caso de prueba: Adquisición de puntos tridimensionales.

A partir de los resultados obtenidos en cuestión de cantidad de puntos tridimensionales y completando el análisis de los resultados mediante pruebas de caja negra, se realizó una clasificación según los datos de entrada, tamaño de SAD y cantidad de puntos salvados por cada uno. (Ver Tabla 15). Si no se obtienen suficientes puntos se debe totalmente al valor de SAD

seleccionado, el mismo según las pruebas que se realizaron deben disminuirse para obtener una mejor calidad visual de la nube de puntos como se muestra en la Figura 34. Para la visualización de la nube de puntos se utilizó el software Rapid Form 2006. (Ver anexo 1).

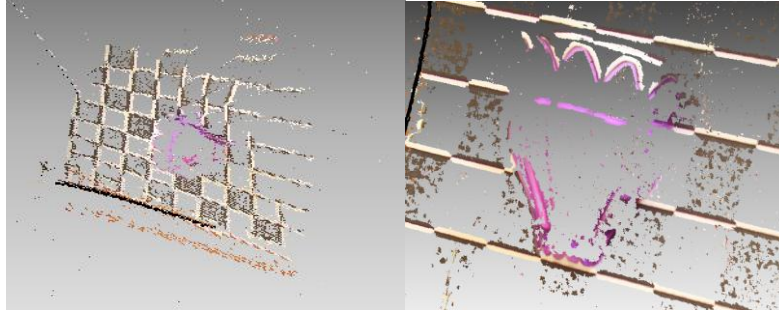


Figura 34. Nube de puntos generada para SAD = 21.

Para completar la validación de la solución, se realizaron encuestas a 12 diseñadores del proyecto Escenarios 3D, pertenecientes a la Línea de Imágenes y Videos basados en Rendering los que representan más del 60% de un total de 20 diseñadores que trabajan directamente en este tema. Los encuestados ofrecieron deferentes opiniones sobre la investigación realizada, en aras de valorar los resultados obtenidos en el prototipo de software. A continuación se recoge un resumen de dichas encuestas y sus preguntas quedan reflejadas en el Anexo 2.

Los resultados arrojados por la encuesta realizada demuestran la solución de la problemática planteada en la línea de diseño. El 100% de los diseñadores a los cuales se les realizó dicha encuesta, encontró beneficios en comenzar a diseñar con una nube inicial de puntos, identificando en un segundo momento, soluciones a algunos de los problemas planteados como optimización de tiempo de modelado y ahorro de materiales y humanos, para un resultados de 100% de todos los encuestados en la identificación de ambas soluciones, estos resultados se reflejan en la Figura 35.

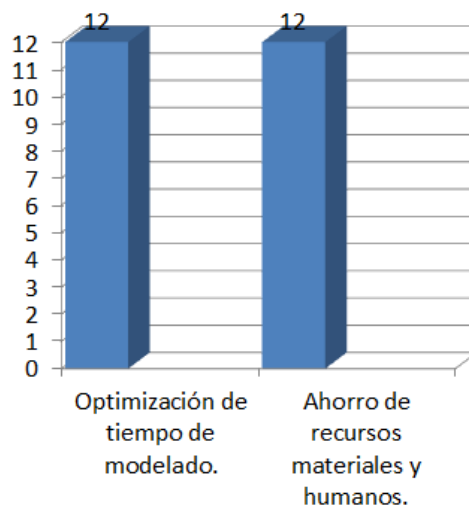


Figura 35. Gráfico generado a partir de los resultados obtenidos de la pregunta dos de la encuesta.

En un tercer momento, valoran los resultados obtenidos por el prototipo desarrollado, la utilización de la nube de puntos para el proceso de modelado, evaluando de 5 el mejor resultado en cuanto a beneficios. Ocho de los diseñadores evaluaron de “bueno” la nube de puntos que se les mostró, representando un 66.6% del total, el resto lo evaluaron de “muy bueno”, que constituye el 33.4%, demostrando así, la aceptación por parte de los diseñadores. En la Figura 36 se muestra el gráfico generado a partir de los resultados obtenidos por la pregunta tres de la encuesta desarrollada.

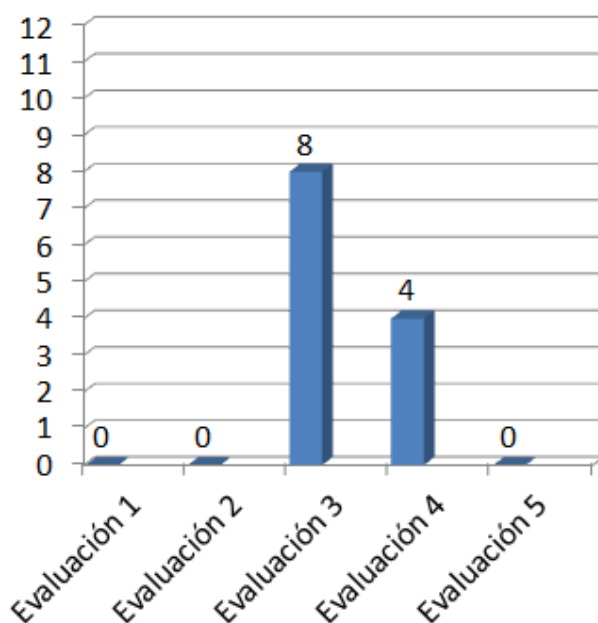


Figura 36. Gráfico generado a partir de los resultados obtenidos de la pregunta tres de la encuesta.

En un último momento, se muestra la importancia de una continuidad en la investigación con el objetivo de mejorar los resultados obtenidos para su utilización en el área del diseño de figuras geométricas, obteniendo también un 100% de aceptación por parte de los diseñadores. El líder del proyecto reflejó también sus opiniones en cuanto a los resultados y la aplicación de un prototipo de esta índole que resuelva algunos de los problemas detectados, en el Anexo 3 se podrá ver dicha opinión.

Por cuanto, la aplicación de este prototipo de software en el proyecto Escenarios 3D se podrá utilizar para optimizar el tiempo de modelado y ahorrar recursos materiales y humanos, quedando demostrada así, su factibilidad.

CONCLUSIONES

Con la realización de este trabajo se profundizó sobre algunos conceptos en el área de la visión por computadora, así como métodos y técnicas eficientes para dar solución al problema planteado. Se llegaron a las siguientes conclusiones:

- ✓ Dentro de los procesos para la adquisición de información tridimensional mediante la fotogrametría, se identificó, la calibración del sistema y la generación automática de puntos como procesos fundamentales en la obtención de gran cantidad de información tridimensional, con una calidad muy aceptable, una gran fiabilidad en la localización de puntos y en la definición de las características geométricas del objeto.
- ✓ Se desarrolló un prototipo de software capaz de adquirir información geométrica de la superficie de un objeto real, arrojando resultados significativos que sirven de punto de partida para el trabajo que se realiza en el proyecto Escenarios 3D, en aras de disminuir el tiempo de modelado y ahorrar recursos humanos y materiales.

RECOMENDACIONES

Este trabajo abrió las puertas a futuras investigaciones, tanto en el área del diseño como en otras áreas de la adquisición tridimensional. A continuación se enuncian algunas recomendaciones orientadas a trabajos futuros:

- ✓ Para la utilización de los resultados se podrán asumir procesos de segmentación y limpieza de la nube de puntos, métodos de modelado y de mallado para obtener un modelo 3D en formato de malla y para los cuales existen programas y algoritmos gratuitos en internet.
- ✓ Para completar o extender el trabajo realizado se podrán añadir técnicas basadas en las características en las que se emplean determinadas estructuras de un nivel superior al píxel para establecer correspondencias entre ellas, las mismas se podrán aplicar para mayor precisión en la obtención de puntos tridimensionales.

BIBLIOGRAFÍA

1. **Nevatia, R.** Machine perception. New Jersey : s.n., 1982.
2. **Martin , Santiago.** Ingeniería Inversa en el diseño para la fabricación aditiva mediante técnicas fotográficas. s.l. : Universidad de Oviedo, Área de Expresión Gráfica en la Ingeniería, 2009.
3. **Young, T. Y. y Fu, K. S.** Hand boook of Patern recongnition and imagen analys Academec Press. New York : s.n., 1986.
4. **Vila Ubiedo, Karen.** Reconstrucción 3D de modelos utilizando técnicas de visión artificial. Madrid : s.n.
5. **Fernandez, E.** <http://www.fing.edu.uy>. [En línea] 2008.
<http://www.fing.edu.uy/ia/deptoggeom/libro/capitulo6/capitulo6.html>.
6. **Rocchini, C.** *A low cost 3D scaner based on structure light*. Madrid : Eurographics, 2006, pág 347.
7. **Gutierrez, S. y Marroquìn, J. R.** *Image and vision computing*. 2004.
8. **Molleda Mere, Julio.** *Técnicas de visión por computador para la reconstrucción en tiempo real de la forma 3D de productos laminados*. Madrid : Tesis Doctoral, 2008.
9. **Jaureque, Luis.** *Introducción a la Fotogrametría*. 2005.
10. **J., Salvi.** *Pattern recognition*. 2002.
11. **Bradske, Bary y Kaebler, Adrian.** *Computer Vision with open library*. O´ Reilly. SAFARY BOOKS.
<https://www.oreilly.com>. Pág 575. 2008.
12. **Jimenez, J. G.** *Visión por Computador*. Madrid : Paraninfo, 1999.
13. **Oviedo, Universidad de.** <https://www.innova.uniovi.es>. [En línea]
<https://www.innova.uniovi.es/innova/campusvirtual/campusvirtual.php>.
14. **Moratinos, Eva.** *Utilización de Técnicas de Ingeniería Inversa para su aplicación en el estudio de Vehículos implicados en Accidentes de Tráfico*. Oviedo : s.n., Marzo 2009.
15. **Martin , Santiago .** *Técnicas Fotográficas de Ingeniería Inversa*. Universidad de Oviedo : s.n. Área de Expresión Gráfica en la Ingeniería, Pág 257, 2009 .
16. **Santiago Martín Laguna, Iñigo Felgueroso Fdez-San-Julián.** *Guía de Diseño de Productos para Rapid Manufacturing*. Universidad de Oviedo : IDEASCAD.2009
17. OBJ Files - A 3D Object Format. [En línea] <http://people.sc.fsu.edu/~jburkardt/data/obj/obj.html>.2008.
18. **Graffigna, Juan Pablo, Romero, Luis Eduardo y R, Raúl.** *Evaluación de métodos para la obtención del mapa de disparidad en sistemas de visión estéreo*. s.l. : Departamento de Electrónica y Automática, U.N.S.J. Pág 16. 2009.

19. **Llorente, César de la Torre , y otros, y otros.** *Arquitectura N-Capas Orientada al Dominio - Microsoft Architecture (1a Edición)*. Noviembre - 2010.
20. **Morales, Frank Hector Rios.** *Arquitectura de Software del Sistema para Graficar Columnas Litológicas de Pozos de Petróleo*. Ciudad de la Habana : s.n, 2010.
21. Metodología de Desarrollo de Software. [En línea] <http://www.scribd.com/doc/12983329/Metodologia-de-Desarrollo-de-Software>.
22. **Giron, Marco Antonio Pereira.** www.scribd.com. [En línea] 2007. [Citado el: 01 de Marzo de 2011.] <http://www.scribd.com/doc/2080534/UML>.
23. **Velazquez, Sergio.** http://www.catalonia.org/cartografia/Clase_07/Fotogrametria/Fotogrametria_index.html. [En línea] Facultat de Nàutica de Barcelona, 2008.
24. **GML Camera Calibration Toolbox, descarga gratuita.** <http://graphics.cs.msu.ru/ru/science/research/calibration/cpp>. [En línea]
25. **tutoriales, Stereo Photo Maker. descarga gratuita con numerosos.** <http://stereo.jpn.org/eng/stphmkr/>. [En línea]. 2008.
26. **PLY - Polygon File Format.** Also known as the Stanford Triangle Format. <http://local.wasp.uwa.edu.au/~pbourke/dataformats/ply/>. [En línea]
27. **White, R y Forsyth, D. A.** *Combining cues: shape from shading and texture. Conference on Computer Vision and Pattern Recognition*. EEUU : s.n., 2006.
28. **Hartley, R. I. y Sturm, P.** *Triangulation. Computer Vision and Image Understanding*. 1997.

ANEXOS

Anexo 1. Transformación de nube de puntos en modelos tridimensionales.

Rapidform es un sistema de modelado de superficies sólido y avanzado para diseñar modelos de CAD paramétricos desde los datos de digitalización 3D.

Esta transformación se realizará con el apoyo de este software gratuito el cual permite ver la verdadera forma del objeto a restituir.

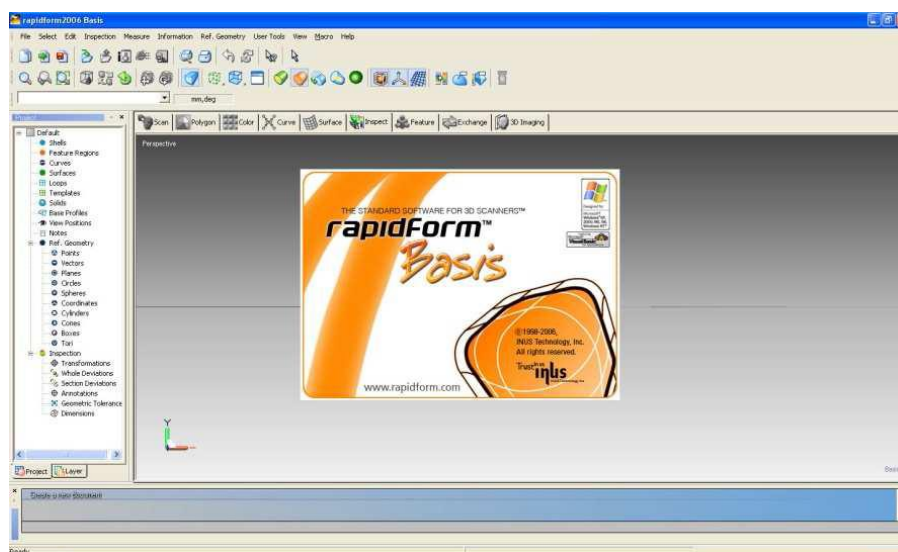


Figura 37. Software de representación de la nube de puntos.

Anexo 2. Encuesta realizada a diseñadores del proyecto Escenarios 3D

1. Sería para usted un beneficio comenzar a diseñar un objeto: (Marque con X una sola opción)
 - a. A partir de cero.
 - b. A partir de una nube de puntos iniciales con información tridimensional del objeto a modelar.

2. Marque con una X, que tipo de beneficios a usted le brindaría.
 - a. Optimización de tiempo de modelado.
 - b. Ahorro de recursos materiales y humanos.

3. Si para usted es un beneficio utilizar una nube de puntos inicial, califique de 1 al 5, el beneficio que le brindaría con la nube de puntos tridimensional que se le mostró. (5 será el mayor beneficio).

_____.

4. Según el resultado apreciado, cree usted importante seguir trabajando en investigaciones de este corte.

_____ Si

_____ No

Anexo 3. Opinión del jefe de la Línea de Imágenes y Videos basados en Rendering.

Considero que el trabajo realizado por el estudiante supera las expectativas por lo que lo catalogo de exitoso. Su aplicación en el proyecto Escenarios 3D se podrá utilizar para optimizar el tiempo de modelado de modo exponencial. Además el avance que reporta el presente trabajo. Es necesario destacar que aunque el resultado es novedoso y útil, se debe seguir trabajando para perfeccionar la captura completa de las zonas muy brillantes que ocasionan agujeros en la nube. De igual forma el pulido de la nube para eliminar escombros de puntos innecesarios, así como el conteo y control de la nube que constituye el objeto en cuestión para reflejar un resultado cuantitativo de optimización en relación con la cantidad total de puntos.

Diosmel Yvonnet Guerra.