

RAE

- 1. TIPO DE DOCUMENTO:** Trabajo de grado para optar por el título de INGENIERO MECATRÓNICO
- 2. TÍTULO:** IMPLEMENTACIÓN DE ALGORITMO DE PROYECCIÓN EN 360° A PARTIR DE IMÁGENES CONTRAPUESTAS
- 3. AUTOR:** Alessandro Fabianno Morelli Torres.
- 4. LUGAR:** Bogotá, D.C
- 5. FECHA:** Julio de 2019
- 6. PALABRAS CLAVE:** 360, Algoritmo, Cámara, Cartografía, Cosido, Equirrectangular, Imagen, Latitud, Longitud, Mecatrónica, Perspectiva, Proyección, Tarjeta Gráfica, Visión.
- 7. DESCRIPCIÓN DEL TRABAJO:** Con el objetivo de implementar un algoritmo capaz de generar una proyección a partir de imágenes capturadas se investigan y desarrollan algoritmos que por medio del uso de proyecciones cartográficas y una esfera de proyección virtual omnidireccional posibiliten el uso de visión en 360° con un sistema de cámaras para producir una imagen única para permitir la máxima visibilidad o un mayor campo de visión en el entorno.
- 8. LÍNEAS DE INVESTIGACION:** Línea de Investigación de la USB: Ingeniería. Sub línea de Facultad de Ingeniería: Procesamiento de señales. Campo Temático del Programa: Procesamiento de señales.
- 9. METODOLOGÍA:** Se plantean 2 enfoques, uno es el enfoque crítico-social en cuanto el problema que se desea solucionar es de esa naturaleza y se plantea el uso de nuevas tecnologías con el fin de mejorar la estructura social, por otro lado, también puede ser tratado como empírico-analítico, en cuanto el objetivo general está directamente relacionado a la interpretación y posible implementación del mundo material en el plano virtual.
- 10. CONCLUSIONES:** En este proyecto se aplica un algoritmo de conversión de proyección de imagen con la finalidad de producir imágenes en 360° en tiempo real. Se implementaron 2 tipos de proyecciones (Equirrectangular y Hammer). La proyección cartográfica empleada, se constituye en un paso intermedio necesario para el tránsito entre proyecciones porque reduce la cantidad de puntos en los que se debe aplicar las ecuaciones de proyección. A pesar de que los algoritmos de proyección pueden implementarse para su uso en tiempo real, existen limitantes externas como el proceso que realiza el firmware de la cámara que retrasan la imagen procesada con respecto a la imagen de origen.

IMPLEMENTACIÓN DE ALGORITMO DE PROYECCIÓN EN 360° A
PARTIR DE IMÁGENES CONTRAPUESTAS

ALESSANDRO MORELLI TORRES

UNIVERSIDAD DE SAN BUENAVENTURA SEDE BOGOTÁ
FACULTAD DE INGENIERÍA
INGENIERÍA MECATRÓNICA
BOGOTÁ D.C.
2019

IMPLEMENTACIÓN DE ALGORITMO DE PROYECCIÓN EN 360° A
PARTIR DE IMÁGENES CONTRAPUESTAS

ALESSANDRO MORELLI TORRES

DIRECTOR: JULIÁN SEVERIANO RODRÍGUEZ

UNIVERSIDAD DE SAN BUENAVENTURA SEDE BOGOTÁ
FACULTAD DE INGENIERÍA
INGENIERÍA MECATRÓNICA
BOGOTÁ D.C.

2019

Dedicatoria

**“I am not what happened to me, I am what
I choose to become”**

- *C. G. Jung*

Dedico esta tesis a mis maestros, quienes me educaron a lo largo de mis estudios, a mis padres, quienes me dieron fuerzas para continuar día tras día, y, sobre todo, a Dios.

Agradecimientos

Agradezco:

A Jai Nankivell, quién me ayudó a lo largo de todo el proyecto con su extensa experiencia en el área y me inspiró creativamente a intentar empujar los límites existentes.

A mi tutor, Julián Severiano, por inspirarme a desarrollarme en el ámbito de la visión artificial y su invaluable ayuda en el proceso de escritura de este documento.

A mis padres, por su abnegado apoyo a lo largo de mi vida y ayudarme a dar lo mejor de mí.

A todos mis maestros de la facultad que me enseñaron a ser un mejor ingeniero.

A Dios, por permitir que todo esto fuera posible.

“If humanism were right in declaring that man is born to be happy, he would not be born to die. Since his body is doomed to die, his task on earth evidently must be of a more spiritual nature. It cannot be the unrestrained enjoyment of everyday life. It cannot be the search for the best ways to obtain material goods and then cheerfully get the most out of them. It has to be the fulfillment of a permanent, earnest duty so that one's life journey may become an experience of moral growth, so that one may leave life a better human being than one started it.”

- Aleksandr Solzhenitsyn

Tabla de Contenidos

1.	Introducción	1
2.	Antecedentes	2
2.1.	Planteamiento del Problema	5
2.2.	Justificación y Pregunta	6
2.3.	Objetivo General	7
2.4.	Objetivos Específicos.....	7
2.5.	Alcances y Limitaciones	7
2.6.	Marco Conceptual.....	8
2.6.1.	Visión.....	8
2.6.2.	Perspectiva	9
2.6.3.	Campo de Visión.....	10
2.6.4.	Visión Artificial	12
2.6.5.	Proyección Cartográfica.....	13
2.7.	Metodología	18
3.	Desarrollo de Ingeniería.....	20
3.1.	Planificación	20
3.1.1.	Selección de cámaras	22
3.1.2.	Selección de proyección	27
3.2.	Desarrollo matemático	29
3.2.1.	Cálculo de proyección.....	29
3.3.	Implementación en tiempo real.....	43
4.	Análisis de Resultados	46

5. Conclusiones	52
6. Recomendaciones	54
7. Bibliografía	55

Lista de Tablas

Tabla 1. Características de distintos arreglos de cámaras..... 27

Lista de figuras

Figura 1. Representación esquemática del proceso de visión en el córtex cerebral humano. Tomado de: (Goodale & Westwood, 2004).....	9
Figura 2. a. Proyección de objeto 3D en pantalla 2D. b. Ejemplo de espacio 3D proyectado. Fuente: Autor.	10
Figura 3. a. Campo de visión monocular (un solo ojo). b. Campo de visión combinado (ambos ojos). (Racette, y otros, 2016).....	11
Figura 4. Diagrama de bloques de un sistema de realidad virtual común. Tomado de: (Zhu , Zhai, & Min, 2018)	13
Figura 5. a. Funcionamiento de una proyección cartográfica. b. Ejemplos de distintas proyecciones. Tomado de: (Snyder & Bugayevskiy, 1995)	13
Figura 6. Ángulos de Latitud y Longitud usualmente utilizados en la representación de la Tierra. (Palmer, 2009).....	14
Figura 7. a. Ejemplo de curvas de nivel. b. Medición topográfica en curvas de nivel de una zona específica. (Heinrichs, y otros, 1995)	15
Figura 8. Proyección equirrectangular del planeta tierra. (Snyder & Voxland, 1989)	16
Figura 9. Ejemplo de proyección equirrectangular. Tomado de: (Gómez, y otros, 2010)	17
Figura 10. Proyección tipo Hammer del planeta tierra. (Grafarend & Syffus, 1997).....	17
Figura 11. Mapa antiguo del planeta Tierra con separación de ojo de pescado.	18
Figura 12. Domo visual que ocupa 360° de visión horizontal. Fuente: Autor.	20
Figura 13. Información en esfera abstracta intercambiable entre proyecciones. Fuente: Autor...	21
Figura 14. Comparación entre espacio virtual (a) y espacio real (b). Fuente: Autor.....	23

Figura 15. a. Aumento del rango de visión con el objetivo de compensar por pérdida de ángulo.	
b. Fusión de campo de visión en ser humano. Fuente: Autor.	23
Figura 16. Imagen sin y con cosido aplicado en la zona intermedia. Fuente: Autor.	24
Figura 17. Representación gráfica de la opción 1 con sus respectivos ángulos de visión. Fuente: Autor.	25
Figura 18. Representación gráfica de la opción 3 con sus respectivos ángulos de visión. Fuente: Autor.	26
Figura 19. Resultados para un escenario interactivo de las proyecciones en distintos grados de campo de visión. (Napieralla, 2018).....	29
Figura 20. a. Ejemplo que exporta la cámara. b. Márgenes de trabajo de FishEye. Fuente: Autor.	30
Figura 21. Cotas determinadas en SolidWorks. Fuente: Autor.	31
Figura 22. Comprobación de proceso de segmentación. Fuente: Autor.	32
Figura 23. Traducción de posición en eje arbitrario de FishEye tipo angular. Fuente: Autor.	33
Figura 24. Posición de P en el espacio abstracto con coordenadas (θ, Φ) . (Bourke, 2016).....	34
Figura 25. Posición de P en el espacio abstracto, en términos de la longitud y la latitud. (Bourke, 2016).....	35
Figura 26. Posición de nueva coordenada P(m,n) en el plano equirrectangular. Fuente: Autor. .	36
Figura 27. Resultado de proyección equirrectangular. Fuente: Autor.	37
Figura 28. a. Segmentos de la imagen a fusionar. b. Atenuación y suma de segmentos en la zona del cosido. Fuente: Autor.	38
Figura 29. Proyección equirrectangular con bordes cosidos. Fuente: Autor.	39
Figura 30. Esfera abstracta proyectada en un espacio tridimensional. Fuente: Autor.	39

Figura 31. Posición de coordenada $P(f,g)$ en el plano Hammer – Aitoff. Fuente: Autor.	40
Figura 32. Resultado de proyección Hammer sin (a) y con el límite aplicado (b). Fuente: Autor.	42
Figura 33. Diagrama de flujo del algoritmo calculado. Fuente: Autor.	43
Figura 34. Desfase en borde de imagen con respecto a límite definido por código. Fuente: Autor.	46
Figura 35. Imagen 1 en proyección equirrectangular con cosido y movida 90° a la izquierda. Fuente: Autor.	47
Figura 36. Imagen 2 en proyección equirrectangular con cosido y movida 90° a la izquierda. Fuente: Autor.	47
Figura 37. Imagen 2 en proyección equirrectangular sin cosido. Fuente: Autor.	48
Figura 38. Imagen 1 en proyección Hammer con cosido y movida 90° a la izquierda. Fuente: Autor.	49
Figura 39. Imagen 2 en proyección Hammer con cosido y movida 90° a la izquierda. Fuente: Autor.....	49

1. Introducción

La visión artificial es una disciplina científica que incluye métodos para adquirir, procesar y analizar imágenes del mundo real con el fin de producir información que pueda ser tratada por una máquina. Se ha ido convirtiendo en una de las más importantes formas de poner en marcha el uso de las nuevas tecnologías al alcance de las personas, incluyendo el campo de la medicina (Dobelle W. , 2000), y en la industria e ingeniería, siendo útil en el control de calidad, el soporte a la producción, la seguridad industrial y el control de procesos (Infaimon, 2018); además, estas técnicas de procesamiento y análisis de imágenes pueden llegar a ser aplicables a diferentes campos de la cotidianidad.

En este proyecto, se reúnen técnicas modernas de la ciencia y la tecnología en visión artificial para generar un campo de visión de 360° a través de una proyección de un espacio real de tres dimensiones en un plano 2d.

Esta proyección tiene tradicionalmente aplicaciones en interfaces gráficas en videojuegos, sin embargo, sus aplicaciones pueden ir desde la robótica para ampliar la percepción del entorno de un robot como en (García & Morales, 2016), hasta la representación del entorno de una persona como en (Zhu , Zhai, & Min, 2018).

En este aspecto, basado en la capacidad del algoritmo de producir distintas proyecciones de 360°, se describe el proceso realizado y se presentan los resultados de la implementación. En un ámbito investigativo, también se tiene el interés de conocer la manera en la cual la máquina procesa la información deseada dependiendo del componente que se encargue de la tarea, donde el tiempo de procesamiento de la imagen es importante debido a que entre menor sea éste, la experiencia del usuario será mejor.

2. Antecedentes

(Brand , Diniz, Ribeiro, & Humayun, 2012) presentan un modelo de visión artificial a través de una estimulación neuronal como prótesis ocular. El artículo expone conceptos biológicos sobre la visión humana, y los aborda desde una perspectiva biomédica presentando alternativas para cubrir ciertas patologías visuales. Describen las ventajas y desventajas de todas las alternativas y técnicas propuestas y concluyen que la visión artificial será de suma importancia e irá mejorando en los próximos años, en especial para pacientes ciegos.

El artículo de (Zhu , Zhai, & Min, 2018) analiza el incremento de usuarios que se está tomando la realidad virtual y destacan su complejidad, ya que depende del comportamiento cognitivo del cerebro. Se enfoca totalmente en la descripción de un campo de visión de 360°, y se afirma que, para tal campo de visión, las imágenes y videos deberían grabar la escena completa del mundo en 3D, y es debido a esto que las resoluciones de imágenes y videos panorámicos son tan altas. Además, se dice que, al mirar un objeto en 360°, el observador puede ver solo una parte en el dispositivo de salida, y por ello renderizar o mostrar la escena completa, es un gasto y desperdicio de recursos. Explican que es posible no malgastar recursos si se puede predecir el actual campo de visión, para enfocarse en renderizar y transmitir la escena específica en ese momento, sin realizar un mapeado completo todo el tiempo. Es necesario desarrollar tales predicciones para procesar las áreas importantes de cada imagen en cada momento, y es una tarea que actualmente no está muy bien adecuada para 360° de campo de visión, porque una imagen en 360° posee características únicas, por ende, en esta investigación se propone estudiar el problema de

la predicción del movimiento de la cabeza, el movimiento del ojo y analizar al observador cuando esté viendo una imagen de 360° en la comodidad de cualquier dispositivo de salida.

En (Chessa, y otros, 2015) se desarrolla un framework inspirado en la visión biológica que integra funcionalidades de ‘bajo nivel de visión’ orientadas a realizar acciones con capacidad de identificación y reconocimiento (realiza un análisis del entorno, lee textos, reconoce objetos y rostros). El objetivo de este framework es realizar un análisis de un video, el cual incluya anotaciones semánticas de la escena 3D e interprete el entorno para evaluar la calidad y discutir las posibles conexiones e integraciones posteriores con el framework en un caso específico.

El documento de (Belbachir, Schraml, Mayerhofer, & Hofstatter, 2018) presenta una cámara de alto rango dinámico (HDR) en tiempo real que muestra una panorámica 3D a 360°. La cámara consta de un par de sensores de visión dinámica que asincrónicamente generan información a una alta velocidad (1 μ s), la cual es transferida por medio de una comunicación gigabit Ethernet, y que están montados en un mecanismo rotatorio de alta velocidad, que puede funcionar hasta por 10 revoluciones por segundo. Se utiliza un algoritmo que mapea la panorámica en una escala de grises mediante un coeficiente de transformación, y luego se reconstruye en tres dimensiones. Como resultado experimental, la cámara es capaz de generar 10 x 3D vistas panorámicas por segundo en tiempo real, en una imagen de 5000x1024 pixeles dentro de un rango dinámico de más de 120 dB bajo iluminación natural.

(Peng, Bennamoun, Wang, Ma, & Zhiyong, 2014) describen que una sola cámara en muy pocas ocasiones provee una cantidad de información satisfactoria debido a su limitado

campo de visión, y proponen dos formas en las que se puede aumentar el campo de visión de una cámara, las cuales son, o usando cámaras omni-direccionales que amplíen directamente el campo de visión, pero la resolución de las imágenes captadas no es uniforme, y como otra alternativa, propone un DAS (Vision Distributed Aperture System), el cual es un montaje de múltiples cámaras fijas para cubrir un gran área de información. Para la implementación de un sistema DAS surgen dos problemas en particular, el primero es la estimación de la orientación de las cámaras en el sistema propuesto y el segundo, la síntesis de una visión virtual en tiempo real. Los resultados que propone el autor demuestran las soluciones para implementar dicho sistema y concluye que las técnicas y métodos usados, son genéricos y son útiles en cualquier caso independientemente del arreglo de cámaras.

La investigación de (Yagi, Imura, Kuroda, & Oshiro, 2011) tiene el objetivo de desarrollar una pantalla de vapor interactiva, la cual permita al observador reconocer la forma en tres dimensiones de un objeto virtual, e interactuar con la imagen mostrada. Los autores indican que, este tipo de pantallas son útiles para proyectar objetos e imágenes flotantes, pero, solamente en dos dimensiones. Basándose en los conceptos de una pantalla de 360° con visión en tres dimensiones, como la Sony's Raymodeler (Ito, 2010), la cual proyecta múltiples imágenes al tiempo, se hace uso de una pantalla cilíndrica de vapor y múltiples proyectores, permitiendo que los objetos proyectados tengan movimiento para los observadores.

El modelado automático en tres dimensiones es un tema que se está desarrollando actualmente, pero la mayoría de software disponible solamente proyecta objetos o

pequeñas escenas de un entorno completo, debido al poco campo de visión disponible en cámaras. En el artículo de (Lhuillier & Nguyen, 2015) se presenta el primer sistema de reconstrucción en tres dimensiones usando cámaras asíncronas sostenidas por un casco, que permita reconstruir grandes áreas o un entorno en su totalidad, y se logró con cámaras de configuraciones similares en su frecuencia, campo de visión y resolución, mediante ajustes y calibración de estos parámetros.

LiveSphere (Kasahara, Nagai , & Rekimoto, 2014) es una tecnología creada a partir del desarrollo de la evolución de experiencias con una inmersión total en tiempo real, que puede ser usado para entretenimiento, deporte, educación, interacción social y asistencia profesional, a través de un arreglo de cámaras que va sobre la cabeza de un sujeto, que provee imágenes esféricas de 360° alrededor del entorno del usuario.

2.1. Planteamiento del Problema

Las limitaciones en el campo visual pueden generar inconvenientes en muchas actividades donde puede desempeñarse un ser humano, por ejemplo, un transeúnte y un motociclista pueden verse inmersos en un accidente al no tener información visual suficiente, este problema evidente ha sido objeto de investigaciones, y es así como los automóviles modernos intentan solucionarlo por medio del uso de espejos o sensores en varios lugares del vehículo.

Por otro lado en el sector defensa, en dónde se requiere un estado constante de alerta se han presentado casos de emboscadas por no tener, de nuevo, un campo de visión completo, lo mismo sucede en el sector de vigilancia, y en general en todas aquellas circunstancias en las que se requiere mayor campo visual, para el que los humanos no estamos diseñados,

caso contrario, por ejemplo, de la mayoría de animales herbívoros (como los ciervos, las vacas y los caballos), los cuales evolucionaron los ojos a los lados (según la hipótesis de Matt Cartmill, 2005) con el objetivo de obtener un mayor campo de visión a cambio de la profundidad (más útil para los animales de caza) debido a la necesidad de poder ver a los depredadores cuando vinieran desde cualquier dirección, esto se suma a otras características de la visión como las pupilas horizontales que captan más luz en ese eje a comparación de las pupilas verticales en los cazadores (Banks, Sprague, Schmoll, Parnell, & Love, 2015).

2.2. Justificación y Pregunta

Un aumento en los ángulos de visión que pueden ser usados por un sensor o un individuo en un determinado proceso en el que se requiera visibilidad, podría mejorar la claridad del proceso y permitir un ahorro considerable en costos y/o en los tiempos de respuesta del sistema en cuestión, sobre todo en el ámbito humano, donde este problema se hace mucho más visible ya que se habla de visibilidad espacial y de poder estar consciente de todo el entorno alrededor del individuo de manera inmediata.

Se plantea desarrollar un proyecto de investigación que permita generar como mecanismo de solución, una herramienta tecnológica de visión omnidireccional. Se requiere diseñar un sistema que posibilite el uso de visión en 360° con un sistema de cámaras para producir una imagen única con el objetivo de permitir la máxima visibilidad o un mayor campo de visión en el entorno. Es así como este proyecto de tesis plantea responder a la pregunta: ¿Cómo implementar un algoritmo de proyección de 360° a partir de múltiples cámaras en tiempo real?

2.3. Objetivo General

Implementar un algoritmo capaz de generar una proyección en 360° a partir de imágenes capturadas.

2.4. Objetivos Específicos

Usar una proyección cartográfica con el fin de mostrar en una esfera visual toda la información proveniente de las cámaras.

Realizar el procesamiento de las imágenes para generar una sola imagen de 360°.

Proyectar las imágenes en vivo sin que exista un retraso significativo entre el tiempo real y la imagen a proyectar.

2.5. Alcances y Limitaciones

Alcances:

- Se programará el sistema en una computadora y no en un sistema embebido.

Limitaciones:

- Las imágenes no son procesadas a la resolución nativa de la cámara.
- El algoritmo funcionará solo en máquinas que usen tarjetas de video Nvidia que puedan compilar en las versiones de OpenGL 3.2 en adelante.
- No se evaluará el nivel de comodidad en los usuarios finales.

2.6. Marco Conceptual

2.6.1. Visión

La visión puede ser considerada como la actividad realizada por sistemas biológicos que perciben distribuciones de luz y construyen una adecuada representación de las escenas en la mente. Dichas representaciones pueden ser re-imaginadas y presentadas en una forma pintoresca, por ejemplo, como un artista describe un paisaje o como se dibuja una imagen en la pantalla de un computador. Hay dos subprocesos/flujos a considerar en dicho método, como lo mencionan (Goodale & Westwood, 2004), el primero siendo el flujo ventral o de la ‘percepción’, el cual gana su nombre debido a que se proyecta desde las áreas visuales tempranas hasta el lóbulo temporal, y proporciona una representación detallada del mundo necesaria para operaciones cognitivas varias como el reconocimiento y la identificación de objetos, usualmente se conoce como la ruta del “qué” (según Chessa, y otros, 2015), y la dorsal, de la ‘acción’ o de la ruta del “dónde”, que va desde las áreas visuales tempranas del cerebro hasta la corteza parietal posterior y provee un control flexible de los módulos visomotores más antiguos con el fin de permitir el control visual de las acciones a realizar y reconoce características como el movimiento y la distancia de los objetos anteriormente reconocidos.

Ambas tareas están organizadas en varias capas jerárquicas de redes neuronales cuyo procesamiento visual se vuelve más informativo y complejo en las capas más altas y trabajan juntas para conformar un solo proceso, la visión (Figura 1), donde la retina envía la imagen a el núcleo geniculado lateral en el tálamo (LGNd), el cual la reenvía a la corteza visual primaria (V1), una vez dentro de la corteza el flujo ventral la proyecta a regiones en

la corteza occipito-temporal y el flujo dorsal a su vez la proyecta a la corteza parietal posterior, el cual a este punto también ha recibido información del colículo superior a través del pulvinar.

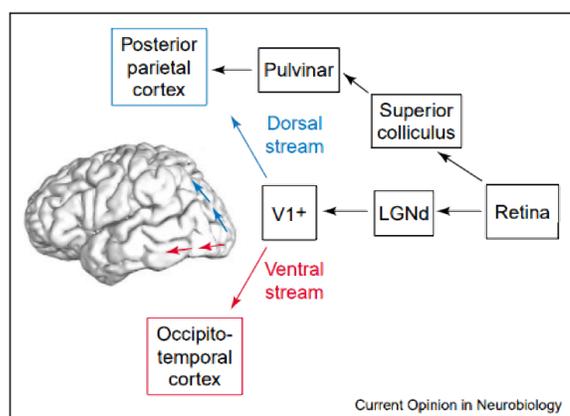


Figura 1. Representación esquemática del proceso de visión en el córtex cerebral humano. Tomado de: (Goodale & Westwood, 2004)

2.6.2. Perspectiva

La perspectiva es la técnica por medio de la cual se busca provocar el efecto de volumen por medio de imágenes planas utilizando la lejanía y la profundidad en un plano 2D (Fig. 2b). Con este fin se usa la perspectiva cónica, tipo de perspectiva que define una serie de líneas y puntos que forman una pirámide, el punto de fuga es particularmente importante y se define como el punto impropio al cual convergen todas las líneas paralelas a una dirección en el espacio, igualmente el punto de vista se define como el punto desde el cual se supone que mira el observador, y coincide con el lugar en el que están nuestros ojos en el espacio (0,0,0 en la Fig. 2a).

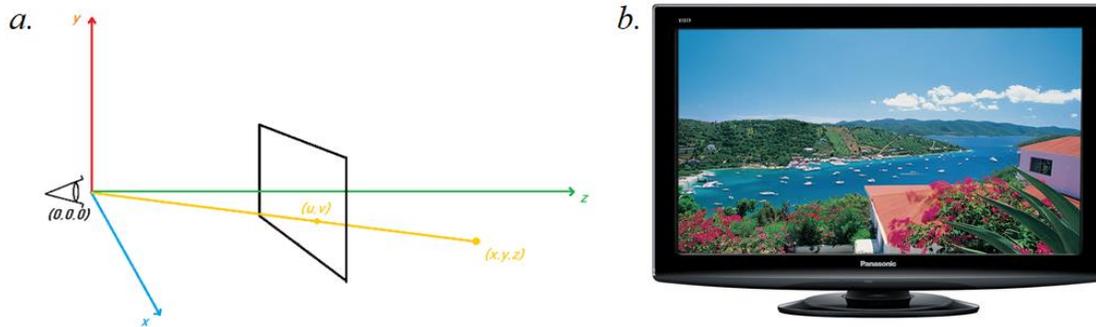


Figura 2. a. Proyección de objeto 3D en pantalla 2D. b. Ejemplo de espacio 3D proyectado. Fuente: Autor.

2.6.3. Campo de Visión

El campo de visión (o FOV) se define como el área en la que los objetos se pueden ver en la visión periférica mientras el centro se mantiene enfocado en un punto específico.

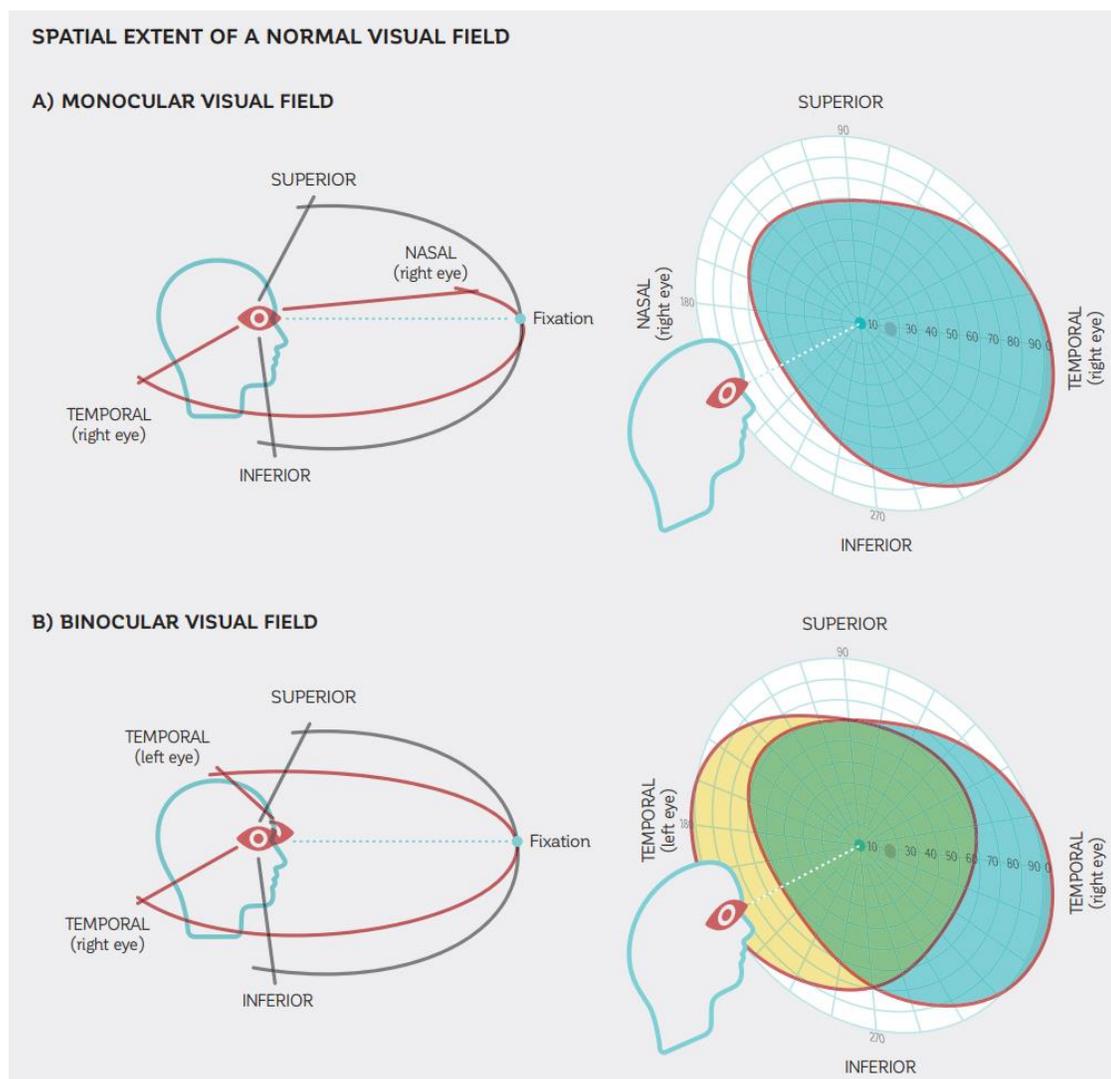


Figura 3. a. Campo de visión monocular (un solo ojo). b. Campo de visión combinado (ambos ojos).

(Racette, y otros, 2016)

Dicho campo de visión usualmente está limitado por la anatomía facial de una persona, la cual varía de un individuo a otro, pero por lo general, la visión en un ojo logra extenderse a los 60° del lado nasal a aproximadamente 90° o tal vez más del lado temporal, y de 60° superiores a 70° inferiores, como mencionan (Racette, y otros, 2016), de esta manera en la visión binocular (Fig. 3b) estos dos campos de visión se integran en la zona intermedia,

borrando la nariz y permitiendo la adquisición de agudeza estereoscópica y profundidad de campo en el sujeto.

2.6.4. Visión Artificial

Tareas como la visión pueden ser realizadas por humanos y demás primates de manera completamente natural, sin embargo, en los últimos años, se diseñaron diferentes sistemas dedicados a ejecutarlas con varios objetivos distintos de acuerdo a su uso, como la mejora de la visión ya existente o el reemplazo de una que ya no funciona, por ejemplo. Para ayudar a solucionar estos problemas existen metodologías y algoritmos que hacen uso de técnicas para representar de diversas maneras el mundo, llamados representaciones artificiales.

Según (Hughes, y otros, 2014) desde la visualización realista de información y la manipulación de las mismas, un gran número de algoritmos han sido desarrollados para mejorar la calidad de la imagen en texturizado y coloreado, control de iluminación, dimensiones y otras áreas; y por ende, hoy la integración de sistemas multimedia, que habilitan el control y la interacción del entorno, permiten generar una realidad virtual del mundo que nos rodea por medio de una cadena de trabajo como la que se puede ver en la imagen de abajo (Figura 4), donde se especifica el set de pasos que se han de realizar con el fin de visualizar información de manera correcta: primero se ha de capturar los datos, luego codificarlos (debido a limitaciones de hardware), enviarlos al segmento donde han de ser trabajados, decodificados, y finalmente, mostrados.

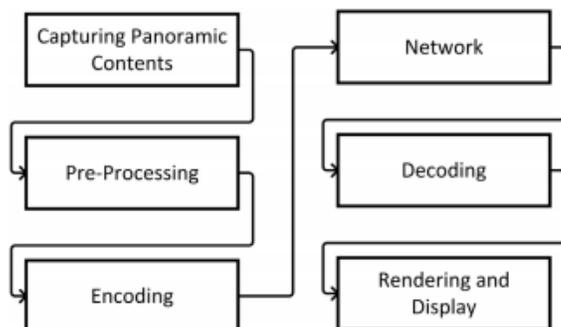


Figura 4. Diagrama de bloques de un sistema de realidad virtual común. Tomado de: (Zhu , Zhai, & Min, 2018)

2.6.5. Proyección Cartográfica

Una proyección cartográfica permite representar cualquier superficie esférica (usualmente utilizada para representar la Tierra) en una lámina plana (Snyder & Bugayevskiy, 1995) y es una representación sistemática de los paralelos y meridianos de un espacio tridimensional en una superficie bidimensional. La figura 5a ilustra el funcionamiento de dicha conversión de la esfera en un plano, donde se usa una ecuación que modela (a modo de lente) distintos puntos en la esfera a puntos en la superficie del mapa, 5b muestra varias proyecciones comúnmente usadas para representar el planeta Tierra a manera de ejemplo.

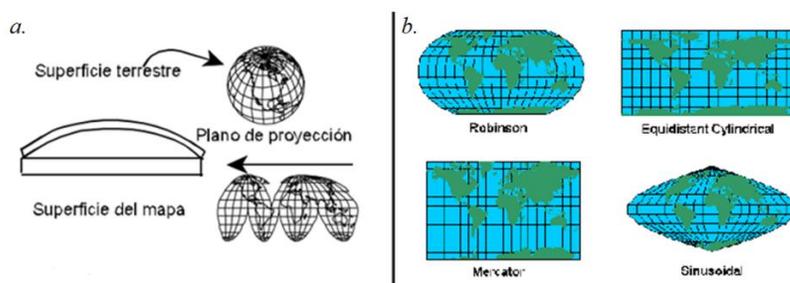


Figura 5. a. Funcionamiento de una proyección cartográfica. b. Ejemplos de distintas proyecciones.

Tomado de: (Snyder & Bugayevskiy, 1995)

En toda representación cartográfica, entonces, la latitud, longitud y altitud son el sistema de coordenadas comúnmente usado para determinar la posición de un punto específico en la superficie de una esfera (Fig. 6). La latitud se calcula con respecto a la línea horizontal en el centro de la esfera llamada el ecuador y se nombra Sur o Norte, dependiendo de la dirección del ángulo que se desee definir (abajo y arriba, respectivamente) siendo su rango desde los 0 a los 90°. La longitud se calcula con respecto a una línea vertical (el meridiano de Greenwich, en el caso del planeta Tierra), y va definida dentro de los 0 a 180° Oeste o Este.

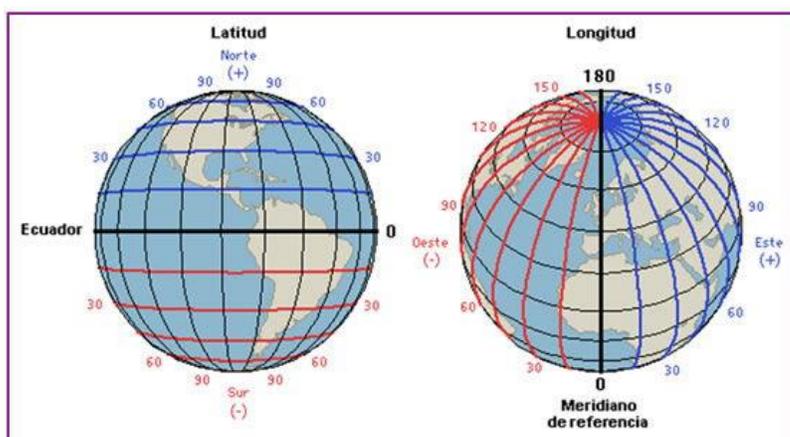


Figura 6. Ángulos de Latitud y Longitud usualmente utilizados en la representación de la Tierra.

(Palmer, 2009)

La altitud en cambio especifica la distancia vertical en un lugar en la superficie y usualmente se mide en metros sobre el nivel del mar (msnm), es debido a esto que en el trazado de los mapas se utiliza un concepto llamado las curvas de nivel (Fig. 7a), las cuales son líneas que unen los puntos de igual altitud, previamente definidos los límites de acuerdo a la frecuencia del muestreo y el rango total a calcular, se calculan basándose en fotografías

aéreas, o de manera más exacta, las medidas topográficas de la zona en cuestión. Distintos tipos de representaciones cartográficas a considerar se detallan en las secciones debajo.

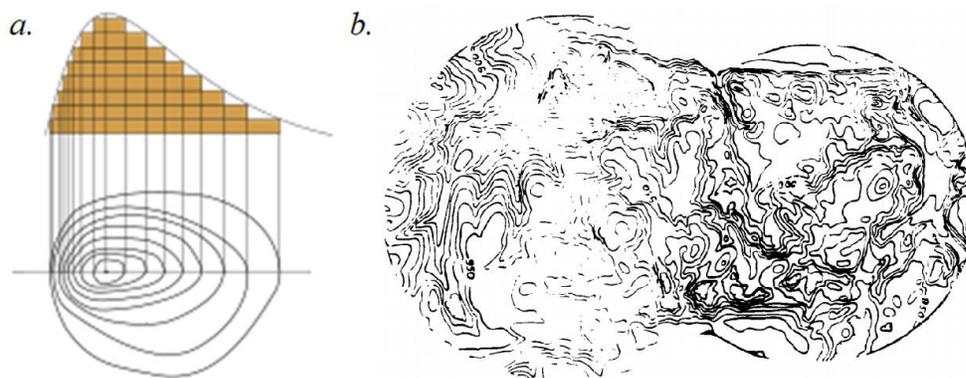


Figura 7. a. Ejemplo de curvas de nivel. b. Medición topográfica en curvas de nivel de una zona específica. (Heinrichs, y otros, 1995)

2.6.5.1. Proyección estereográfica o equirrectangular

En esta proyección, la coordenada horizontal y la vertical se mapean 1 a 1 con la longitud, y la latitud (como se puede observar en la figura 8) y por lo tanto posee una transformación o escalado aplicados. En esta proyección, el sistema de coordenadas no se determina geoméricamente, sino por una mera división empírica del plano, no obstante, es el origen de la teoría de proyecciones cartográficas. En ella, todos los paralelos se representan en la misma magnitud que un punto de referencia, de donde la deformación en sentido longitudinal aumenta conforme la latitud. Debido a la sencillez de sus cálculos, su uso fue más común en el pasado.

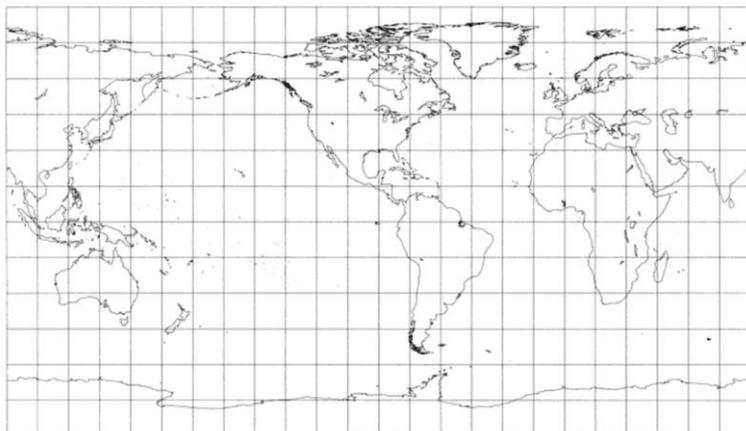


Figura 8. Proyección equirrectangular del planeta tierra. (Snyder & Voxland, 1989)

La proyección equirrectangular es habitualmente empleada para las imágenes esféricas, y consiste en imponer que cualquier sector esférico abarcado por ángulos iguales a lo largo de dos paralelos y dos meridianos se represente mediante un cuadrado. Las imágenes o figuras que usan este tipo de proyección también se conocen como panorámicas (Figura 9), y es un formato muy común y utilizado en entornos actuales, ya que, todos los cuadrados de la retícula corresponden con sectores esféricos de igual longitud y latitud. De esta manera, representado en la figura, los elementos rectilíneos verticales de la realidad se conservan, pero no los elementos rectilíneos horizontales, es decir que todas las verticales permanecen verticales, y el horizonte se convierte en una línea recta a través del centro de la imagen. (Gómez, y otros, 2010)

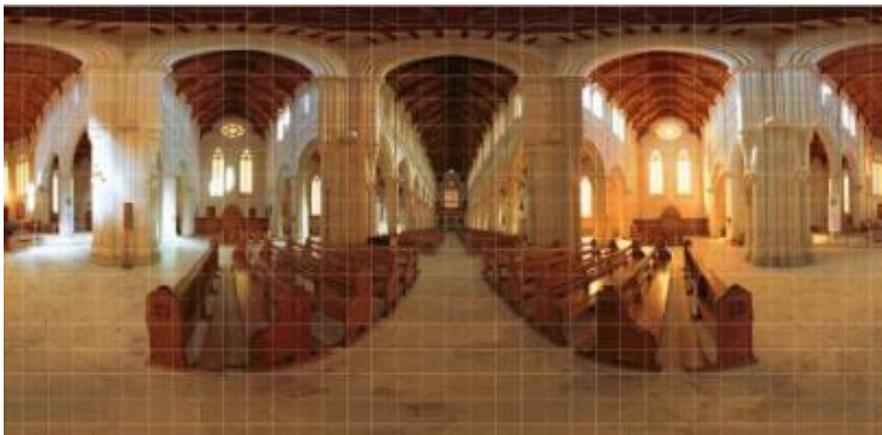


Figura 9. Ejemplo de proyección equirrectangular. Tomado de: (Gómez, y otros, 2010)

2.6.5.2. Proyección tipo Hammer

La proyección tipo Hammer se puede definir según (Tomás, Ferreiro, Sentana, & Díaz, 2002) como un tipo de proyección que define una inversión en el espacio que transforma los puntos de la esfera en puntos del plano. Es un tipo de proyección conforme, es decir, que conserva la verdadera magnitud de las distancias en la proyección. (Fig. 10)

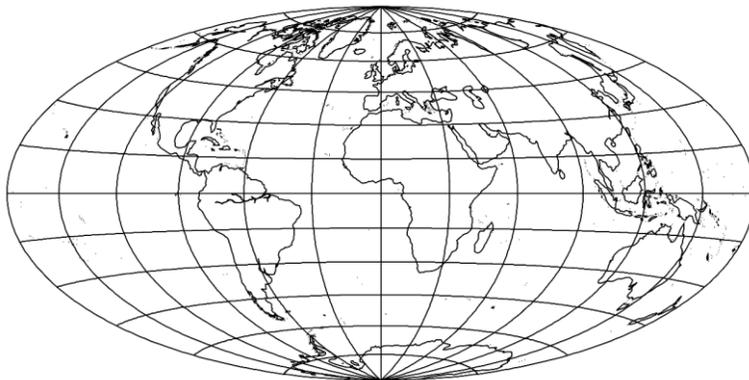


Figura 10. Proyección tipo Hammer del planeta tierra. (Grafarend & Syffus, 1997)

2.6.5.3. Proyección FishEye o de ojo de pescado

Otra proyección existente que se ha de tener en cuenta en el proceso de cálculo es la proyección en ojo de pescado de tipo angular (Fig. 11), que se define de manera tal que la

relación entre las distancias tanto en el centro como acercándose a los extremos sea siempre la misma, y aunque puede mantener los 360° de visión en un solo lente, en este caso se utilizan 2 ojos de pescado de 180° cada uno debido a la naturaleza de la cámara a usar.

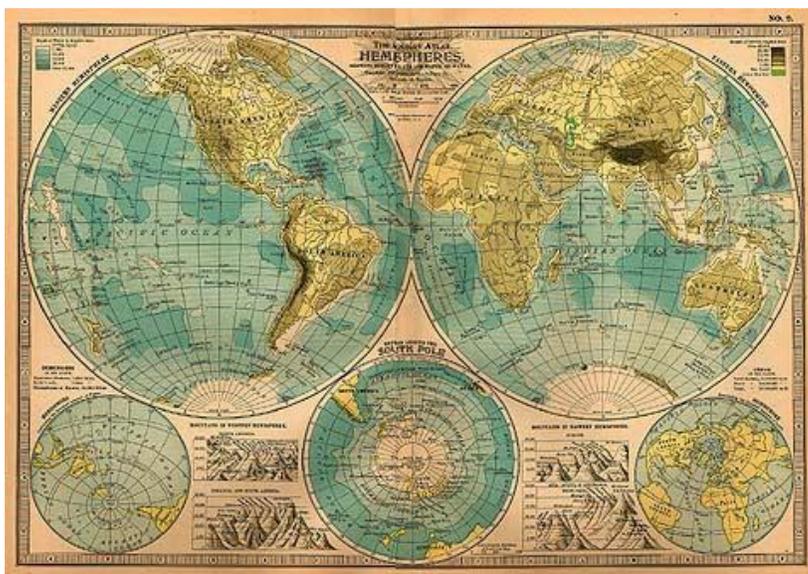


Figura 11. Mapa antiguo del planeta Tierra con separación de ojo de pescado.

2.7. Metodología

De acuerdo a la justificación del proyecto, se plantean 2 enfoques, uno es el enfoque crítico-social en cuanto el problema que se desea solucionar es de esa naturaleza y se plantea el uso de nuevas tecnologías con el fin de mejorar la estructura social, por otro lado, también puede ser tratado como empírico-analítico, en cuanto el objetivo general está directamente relacionado a la interpretación y posible implementación del mundo material en el plano virtual.

Esta tesis es por lo tanto una investigación aplicada, ya que su propósito es dar solución a una situación en concreto, en virtud de utilizar técnicas para el procesamiento de

imágenes, basadas en conocimientos de sistemas de visión artificial y congrega todos los rasgos detallados de un nivel exploratorio donde se permita señalar la calidad del problema, correlacionado y descriptivo.

Se plantea el logro de los objetivos empezando por una fase de planificación de un mes en la cual se definirán los materiales necesarios por medio de un proceso comparativo de acuerdo a sus características y el marco de referencia dentro del cual se trabajará de acuerdo al tiempo disponible, en la siguiente fase, puramente matemática, se definirá las ecuaciones pertinentes al algoritmo en un nivel alto de abstracción, para que luego, por medio de la implementación del código en un lenguaje de programación robusto se pueda cumplir con las especificaciones planteadas en una ventana de tiempo de 4 meses. Sin embargo, testear las características del algoritmo en cuestión directamente en dicho lenguaje se haría un proceso demasiado tedioso, por lo cual se plantea una tercera fase en la cual es pertinente el uso de un lenguaje intermedio de más alto nivel en el cual verificar la validez de dichas ecuaciones, debido a la experticia del autor en el software MatLab y su facilidad a la hora de corregir errores de programación se elige esta interfaz. Una vez implementado correctamente se emplea un proceso iterativo para encontrar los valores más efectivos en las variables que queden libres luego de la investigación de la primera fase, las cuales dependerán de los materiales físicos a usar. A partir de aquí se pasa a la fase 4, donde se programa y testea el código final en el lenguaje de confianza y se logran todos los objetivos. Para concluir se detallarán los resultados e inconveniencias encontradas en cada fase del desarrollo y las recomendaciones para futuras mejoras posibles en el proyecto.

3. Desarrollo de Ingeniería

3.1. Planificación

Al tipo de visión que tenemos los seres humanos se le conoce como **Vista de Perspectiva** (ver sección 2.6.2), en el cual ninguna proyección ha sido aplicada y por medio del cual percibimos el mundo que nos rodea por medio de nuestros propios ojos, sin embargo, al ser usado virtualmente, presenta una serie de problemas como la deformación que existe al ver una pantalla desde el punto de vista incorrecto o su campo de visión, que está limitado al campo de visión que la pantalla cubra de la visión total disponible del usuario.

Varias soluciones existen para los problemas mencionados arriba, como los cascos de Realidad Virtual, por ejemplo, donde los últimos modelos disponibles en el mercado ofrecen 145° para el HTC Vive, 120° para el Oculus Rift y cerca de 100° para el Samsung Gear VR (todos medidos en diagonal, respectivamente), (Hunt, 2018). Se plantea, entonces, en este documento, la implementación de un sistema en 360° que logre ocupar todos los ángulos de visión existentes, como se observa en la figura 12.



Figura 12. Domo visual que ocupa 360° de visión horizontal. Fuente: Autor.

Para pasar de un sistema en 360° a otro se plantea utilizar un globo abstracto, esto es, una superficie esférica que tiene como centro y origen del sistema coordinado al usuario y que permita modelar las direcciones en cuestión, ya no en términos de las coordenadas de imagen (u,v) (ver Fig. 2), sino en la latitud y la longitud de dicha superficie, la cual al rodear al usuario ahora permite usarla de intermediario entre un sistema coordinado y otro, en este caso en particular la altitud de dicha esfera se obvia debido a que trabajamos en una esfera lisa de magnitud unitaria (el globo abstracto posee diferencial de altitud 0 en todos sus puntos), y por lo tanto trabajamos entonces solo con el sistema de coordenadas angular (*longitud,latitud*).

Cada proyección distinta tiene una serie de ecuaciones que se puede usar para modelar la posición en (u,v) de un pixel en la pantalla por medio de una correspondiente latitud y longitud en el globo abstracto, y por lo tanto, con la aplicación de dichas ecuaciones la información puede ser intercambiada entre todas las proyecciones disponibles de manera más sencilla (Fig. 13).

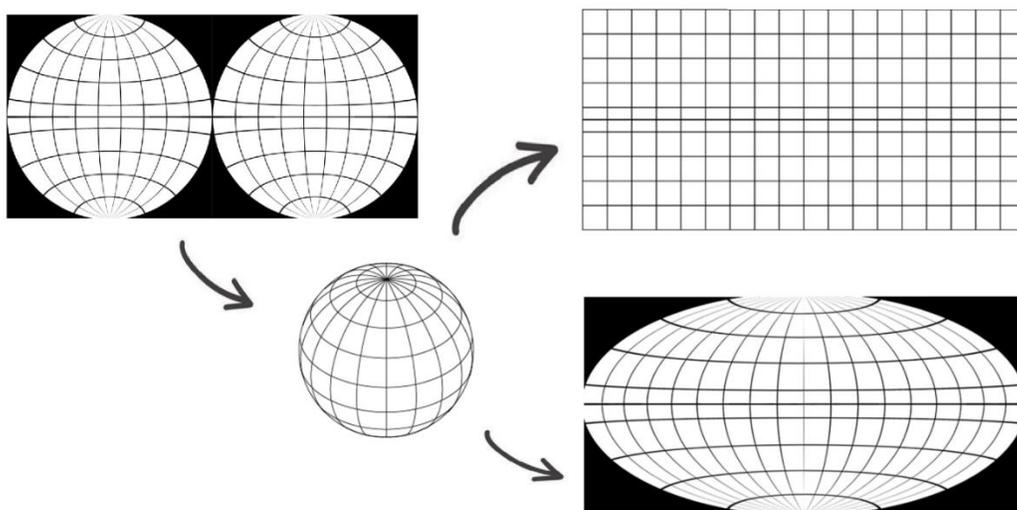


Figura 13. Información en esfera abstracta intercambiable entre proyecciones. Fuente: Autor.

3.1.1. Selección de cámaras

Para poder realizar la proyección que permita obtener una imagen en 360° a partir del espacio abstracto, es necesario capturar las imágenes del entorno. En un ambiente virtual es válido suponer que la información de la cámara se captura desde un punto origen sin dimensiones (Fig. 14a); sin embargo, en la vida real existe un inconveniente, las cámaras, el armazón y/o el circuito que se pueda montar, ocupan todos un espacio que separa los lentes entre sí de su posición ideal, como se ve en la figura 14b, donde el espacio ahora usado para componentes se marca con “NN”, y como producto de este desplazamiento el ángulo de visión de los lentes ya no es suficiente para cubrir el espacio completo, quedando una zona ciega (marcada en la imagen como una zona gris) en la cual no se puede ver nada debido a que ninguna cámara puede recibir ahora dicha información, las únicas maneras de corregir ese problema, entonces, son, intentar acercar las cámaras entre sí, agregando más cámaras en el proceso para cubrir el mismo espacio, cosa que resulta costosa, o intercambiar las cámaras por otras con un mayor ángulo de visión, como se puede ver en la figura 15a, disminuyendo la cantidad de espacio ciego, donde fuera del círculo se vuelve a cubrir todo el rango necesario para la esfera abstracta.

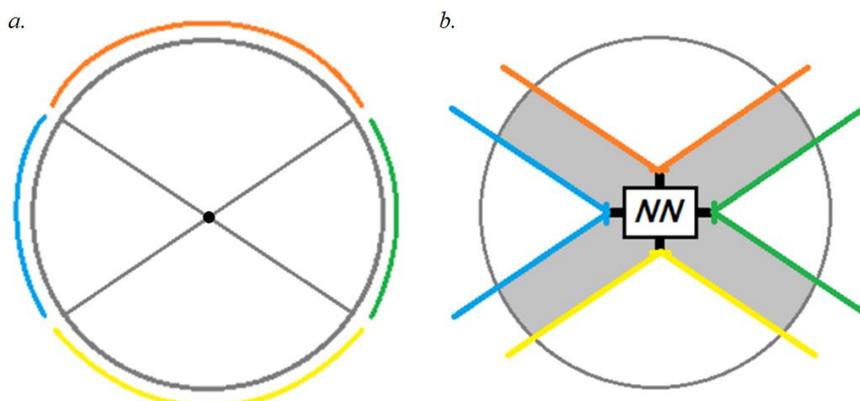


Figura 14. Comparación entre espacio virtual (a) y espacio real (b). Fuente: Autor.

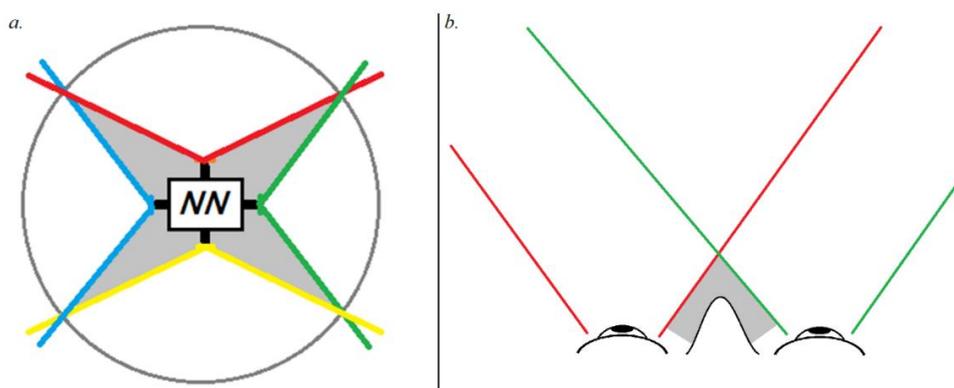


Figura 15. a. Aumento del rango de visión con el objetivo de compensar por pérdida de ángulo. b.

Fusión de campo de visión en ser humano. Fuente: Autor.

Esta reunión de los fragmentos antes desperdiciados en la separación produce un efecto de visión doble en la zona de la unión, dicho efecto es idéntico al producido por los ojos humanos en la zona intermedia (fig. 15b) (Nagata, 1996) donde las imágenes capturadas por ambos conos de dos lentes distintos se perciben ligeramente diferente debido a que son un segmento de la imagen donde cada lente envía una imagen distinta ya que la información no está siendo capturada desde un mismo punto de origen; con el fin de corregir este problema se utiliza un método llamado “cosido” (como se muestra en la figura 16b), cuyo

objetivo es, según (Levin, Zomet, Peleg, & Weiss, 2004): “producir un mosaico visualmente plausible con dos propiedades deseables: primero, el mosaico debe ser lo más similar posible a las imágenes de entrada, tanto geométrica como fotométricamente. En segundo lugar, la costura entre las imágenes cosidas debe ser invisible”.



Figura 16. Imagen sin y con cosido aplicado en la zona intermedia. Fuente: Autor.

Se revisan entonces varias alternativas de sensores (o cámaras) que puedan capturar dicha información, y debido a que la mayoría de las cámaras presentes en el mercado actual no ofrece 360° de visión por sí solas es necesario considerar diferentes arreglos de acuerdo al modelo que permitan trabajar el campo de visión deseado. El criterio que se usa para elegir el sistema correcto varía con respecto a determinadas características técnicas de las alternativas presentes en el modelo que son comparadas entre sí de acuerdo con su conveniencia para el proyecto. A continuación, se hace una revisión de las opciones disponibles para conseguir el procesamiento de imagen deseado.

Opción 1: Arreglo de 6 cámaras de 100°

Como primera opción se considera cubrir el espacio abstracto con 6 cámaras posicionadas en el lado positivo y negativo de cada uno de los 3 ejes de un sistema cartesiano tridimensional (Fig. 17), este es el modelo que más cosido presenta debido a que posee el mayor número de aristas y es también la que posee el mayor volumen de espacio sin usar (el volumen total interno por debajo de las cámaras con información que no puede ser capturada por ninguna de ellas), para esta opción se consideró la cámara modelo “SJCAM M20”. (Mostrada en la Tabla 1)

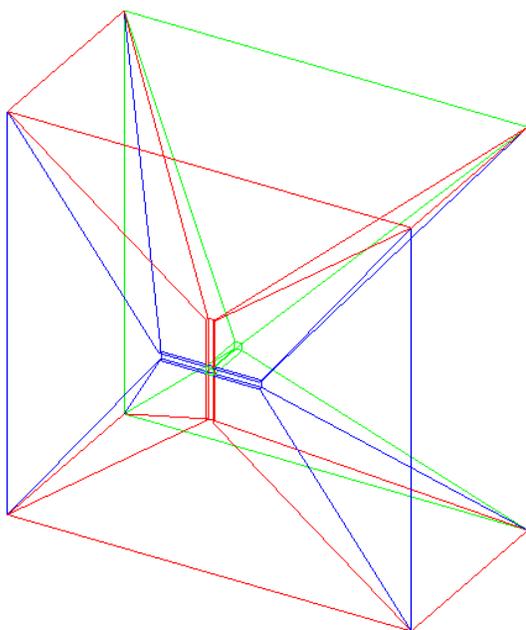


Figura 17. Representación gráfica de la opción 1 con sus ángulos de visión. Fuente: Autor.

Opción 2: Arreglo de 4 cámaras de 140°

La segunda opción es ubicar 4 cámaras con 140° de visión en forma de tetraedro, esta es la figura que provee el menor volumen de espacio posible debido a que es la forma con menos vértices posible de hacer usando ángulos rectos, para esta opción se consideró la cámara modelo “HA-I01W-20”. (Mostrada en la Tabla 1)

Opción 3: Arreglo de 2 cámaras de 180°

Es también posible usar 2 cámaras de 180° para formar la esfera (Fig.18), esta opción presenta la menor cantidad de cosidos posibles (1 solo) y puede tener los menores costos, para esta opción se consideró la cámara modelo “360Fly 4K - H”. (Mostrada en la Tabla 1)

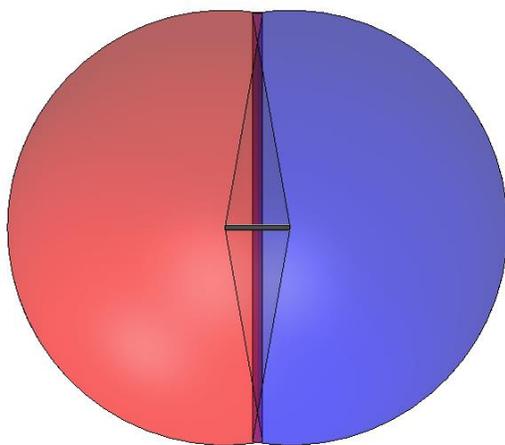


Figura 18. Representación gráfica de la opción 3 con sus respectivos ángulos de visión. Fuente:

Autor.

Opción 4: Cámara de 360° con dúo de lentes de 190°

En último lugar se considera usar una única cámara de 360°, cuya única diferencia con la opción anterior es que los dos lentes no tienen separación móvil entre sí, y vienen en su propio armazón, minimizando así costos de construcción, aquí también se presenta cosido debido a que el número de lentes es igual, para esta opción se consideró la cámara modelo “Xiaomi Mijia MI Sphere”. (Mostrada en la Tabla 1)

La tabla 1 muestra las características clave de las 4 opciones a sopesar entre sí, estas siendo su tamaño, para tener en cuenta cómo ha de afectar el cosido de la imagen final, la

resolución, clave debido a que a mayor resolución mayor detalle en la imagen, su tasa de actualización y su precio.

Tabla 1. Características de distintos arreglos de cámaras.

Opción	Características de las opciones			Cuadros por segundo
	Dimensiones (cm)	Resolución	Precio (pesos)	
SJCAM M20 x6	5x4x2.9	2k	\$2'332.116	60
HA-I01W-20 x4	6x3x4	2k	\$1'416.880	30
360Fly 4K – H x2	6x9x3.5	4k	\$833.464	60
Mijia MI Sphere	8x7x1.26	4k	\$792.153	60

De las 4 cámaras evaluadas se descarta la segunda debido a que su velocidad es demasiado lenta para el correcto análisis de la información, como las características técnicas de los 3 restantes son adecuadas para el desarrollo del proyecto, se emplea la “MIJIA MI SPHERE” debido a que posee una buena resolución, sus dimensiones son apropiadas y se adecúa mejor al presupuesto del mismo.

3.1.2. Selección de proyección

Con la cámara a usar seleccionada y comprada se procede a seleccionar las proyecciones que se programa con base en los problemas que se desea solucionar, a continuación se describen las proyecciones seleccionadas así como la forma de implementarlas.

3.1.2.1. Criterio de selección

El tipo de proyección empleado en este proyecto se selecciona basándose en un informe de grado realizado en el *Blekinge Institute of Technology* (Napieralla, 2018), el cual tiene como propósito investigar qué proyección es la adecuada para cada entorno de acuerdo a la calidad percibida de distintos medios con distintos grados de interacción con el usuario

(pruebas fotográficas, cinematográficos e interactivos, esto no solo debido a que se responde distinto a imágenes estáticas, en movimiento e interactivas, sino que todas son relevantes gracias a que todas representan un medio a través del cual la información podría llegar a ser percibida en ese formato), en distintos grados de campo de visión (110°, 140° y 170°) y usando 3 distintas proyecciones (de perspectiva, Panini y Equirrectangular) y cuyo resultado final se obtiene mediante la aplicación de una escala de Likert (Likert, 1932), midiendo el nivel de confort o naturalidad que se percibe en la proyección con la configuración que había sido previamente mostrada, y donde se pedía calificar la prueba de acuerdo a “Qué tan natural te parece la proyección actual que estás viendo?” del 1 al 5, siendo 1 (muy antinatural) y 5 (muy natural).

Los resultados medidos demuestran una caída en la calidad percibida a un ritmo exponencial en todas las proyecciones (fig. 19), sin embargo, la vista en perspectiva decae mucho más rápidamente al aumentar el campo de visión que las otras proyecciones, las cuales, en cambio, mientras obtienen una mejor puntuación en altos grados de visión se notan menos consistentes y bastante dependientes del entorno (resultando así en una mucho mayor desviación estándar).

En la figura se pueden ver los promedios de calificaciones y sus desviaciones para cada combinación de perspectiva y campo de visión posible, mostrando las 3 distintas perspectivas en el eje horizontal, las calificaciones en el eje vertical y los campos en distintos colores.

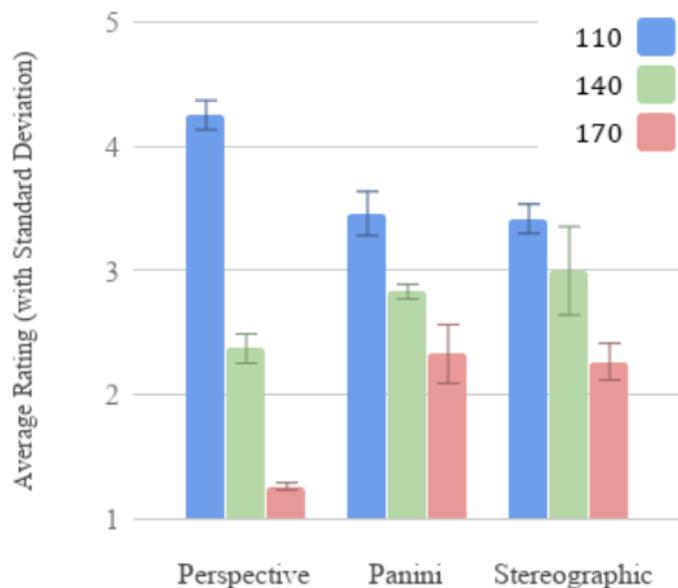


Figura 19. Resultados para un escenario interactivo de las proyecciones en distintos grados de campo de visión. (Napieralla, 2018)

Teniendo en cuenta la anterior información, se decide que se ha de seleccionar la proyección equirrectangular porque es la que menor variación presenta en el nivel de comodidad con los cambios en el campo de visión. Además, se implementa en este trabajo la proyección de Hammer-Aitoff.

3.2. Desarrollo matemático

3.2.1. Cálculo de proyección

Partiendo de las dos imágenes semiesféricas (definidas en 2.6.5.3) complementarias con formato FishEye que exporta la cámara seleccionada (ver sección 3.1.1), se procede entonces a la implementación de las proyecciones seleccionadas mediante la aplicación de las ecuaciones correspondientes (Ho & Budagavi, 2017).

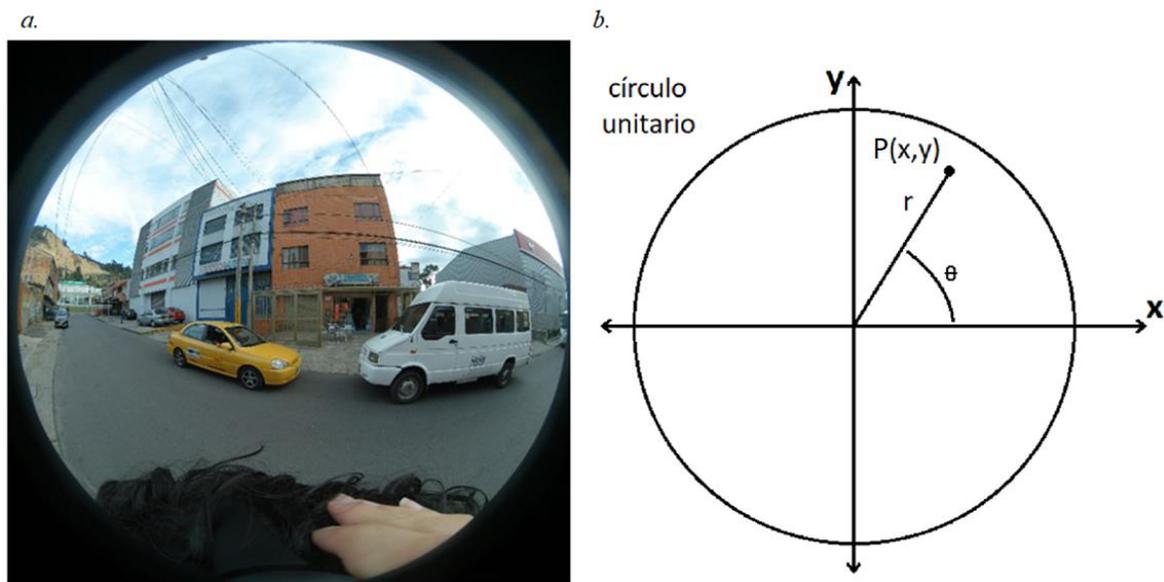


Figura 20. a. Ejemplo que exporta la cámara. b. Márgenes de trabajo de FishEye. Fuente: Autor.

Es necesario pasar la imagen presente en el FishEye al espacio abstracto, de manera que se empieza por definir los márgenes de trabajo en el plano del FishEye representado por la figura 20b, siendo el punto P con coordenadas (x,y) la ubicación de un pixel determinado, r el radio desde el punto de origen hasta P y θ el ángulo existente entre dicha recta y el eje X, que existen dentro de un margen unitario (de -1 a 1 en todas las direcciones).

Se programa un segmento que extrae los fragmentos de FishEye de la imagen, con este objetivo se permite ingresar una serie de 3 dimensiones que definirán las características del corte: el radio de los círculos y su distancia en X y Y con respecto al centro. Para determinar los valores de los bordes a usar en la imagen se usa el programa SolidWorks (programa de diseño de piezas con uso ingenieril) como se ve en la Figura 21 debido a su fácil uso y el manejo de magnitudes escalares precisas en sus acotaciones.

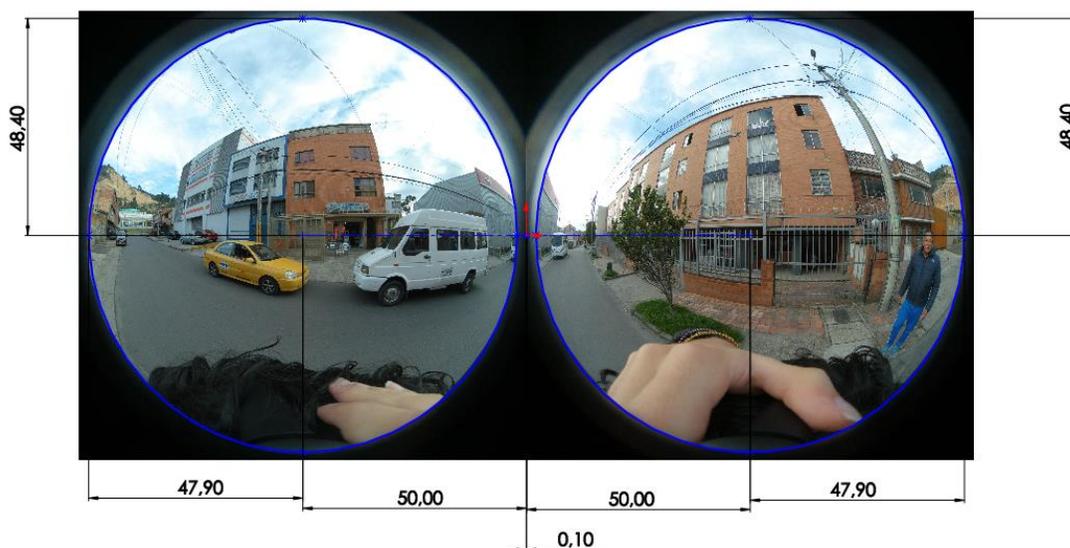


Figura 21. Cotas determinadas en SolidWorks. Fuente: Autor.

Una vez completado el proceso de acotado (se maneja en la Fig. 21 números ampliados en una escala de 100 a 1 con el objetivo de obtener más puntos decimales), se nota que las zonas a extraer en la imagen, más que círculos son elipses (característica que puede ser particular a el producto utilizado en el diseño), por lo tanto se nota que se requieren datos adicionales para el cálculo del área, y se definen las elipses de acuerdo a su radio en X y en Y, también se nota un pequeño desvío presente con respecto al centro tanto en el eje X como en el eje Y, y se tienen estos valores también en cuenta, el resultado de la aplicación de los bordes se puede ver en la figura 22.

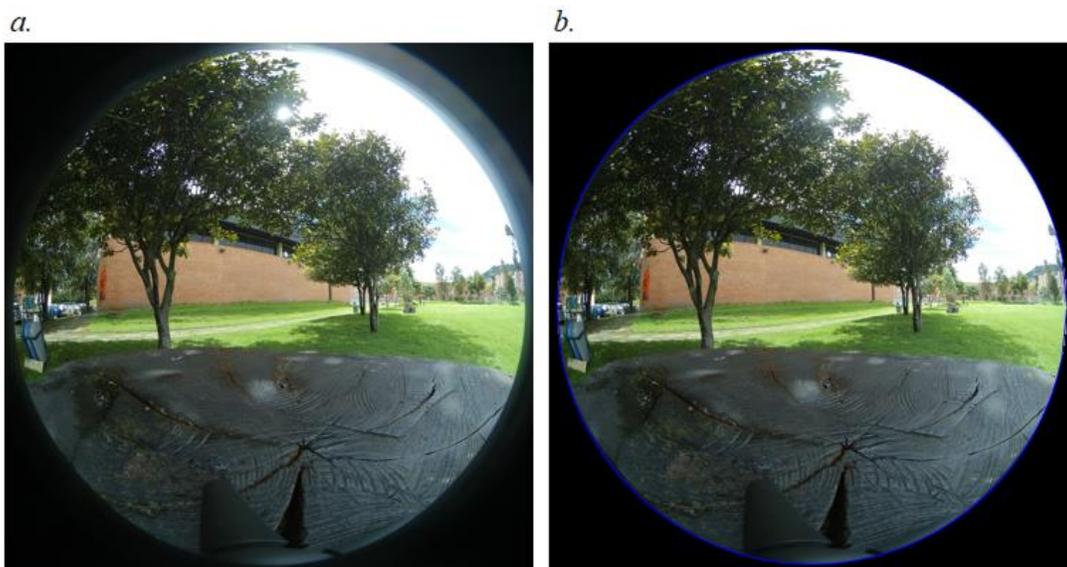


Figura 22. Comprobación de proceso de segmentación. Fuente: Autor.

Posteriormente las imágenes se redimensionan para ajustarlas al tamaño de círculos perfectos, mientras este paso no es necesario si la cámara en cuestión ya los exporta de esa manera, lo más probable es que este no sea el caso y exista una desviación de al menos un píxel.

La cámara usa un lente FishEye de tipo angular (Fig. 23), que se define como la proyección donde la cual la distancia desde el centro de la imagen es proporcional al ángulo de avance de la cámara en cuestión, debido a que dicha relación es constante, este tipo de lente permite la proyección de ángulos mayores a 180° , y la información está igualmente distribuida en todas las direcciones (en términos de magnitud). De esta manera, la información se ve proyectada en un plano arbitrariamente designado en una relación directa entre la dirección a tomar y la magnitud del ángulo que refleja en la esfera abstracta.

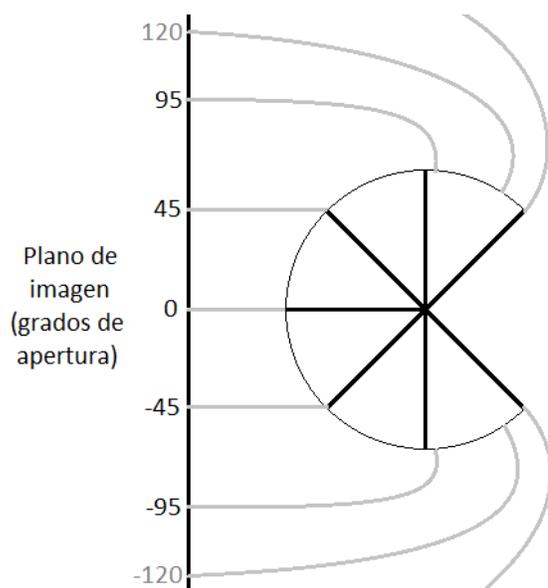


Figura 23. Traducción de posición en eje arbitrario de FishEye tipo angular. Fuente: Autor.

Otro elemento para considerar es la apertura de los lentes presentes en la cámara, la cual representa el campo angular de visión del FishEye y depende de la forma del lente, y como no todos los lentes son exactamente iguales debido a condiciones particulares de diseño o posibles defectos en la construcción, se puede presentar una pequeña variación en sus características, el manual del usuario de la cámara empleada afirma que hay $\sim 190^\circ$ de apertura en cada lente. Teniendo en cuenta el tipo del lente y su apertura, se definen las ecuaciones que modelan los ángulos en el espacio abstracto dependiendo de la posición (Fig. 24) como:

$$\Phi = \frac{r * ap}{2} \quad (1)$$

$$\theta = atan2(P_y, P_x) \quad (2)$$

Siendo r el radio desde el punto de origen hasta el punto que se está calculando, ap la apertura del lente, θ el ángulo existente entre dicha recta y el eje X, Φ el ángulo al eje Y, y $atan2$ la tangente inversa de cuatro cuadrantes, que se define como la ecuación 3.

$$atan2(y, x) = \begin{cases} \arctan\left(\frac{y}{x}\right) & \text{if } x > 0, \\ \frac{\pi}{2} - \arctan\left(\frac{x}{y}\right) & \text{if } y > 0, \\ -\frac{\pi}{2} - \arctan\left(\frac{x}{y}\right) & \text{if } y < 0, \\ \arctan\left(\frac{y}{x}\right) \pm \pi & \text{if } x < 0, \\ \text{undefined} & \text{if } x = 0 \text{ and } y = 0. \end{cases} \quad (3)$$

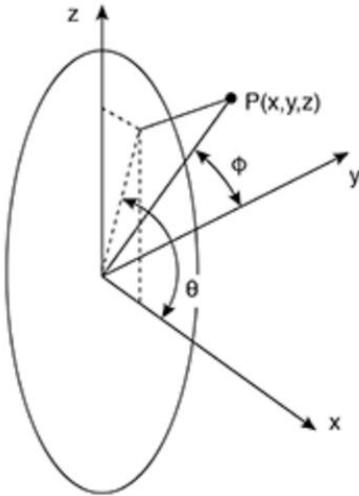


Figura 24. Posición de P en el espacio abstracto con coordenadas (θ, Φ) . (Bourke, 2016)

Los ángulos calculados son relativos al plano XZ (el cual contiene la información capturada del FishEye) y necesitan ser convertidos a coordenadas de latitud y longitud de la siguiente manera. La figura 25 ilustra el punto P en términos de dichas coordenadas.

$$long = atan2(P_y, P_x) \quad (4)$$

$$lat = atan2(P_z, \sqrt{P_x^2 + P_y^2}) \quad (5)$$

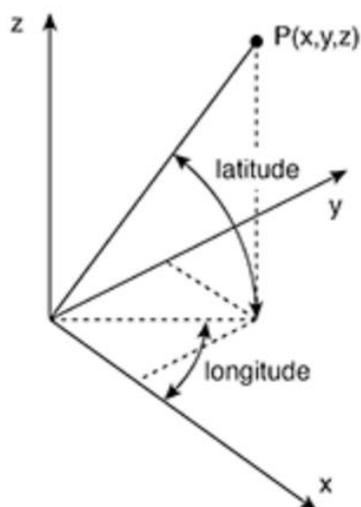


Figura 25. Posición de P en el espacio abstracto, en términos de la longitud y la latitud. (Bourke, 2016)

3.2.1.1. Cálculo de proyección Equirrectangular

A la proyección equirrectangular también se le suele llamar proyección esférica, el motivo es que es una representación 1:1 de la esfera abstracta, donde la longitud se traduce directamente en la nueva coordenada x y la latitud en la nueva coordenada y. Su implementación se describe a continuación:

Siendo *sicir* (*size of circle*, o tamaño del círculo) la altura en píxeles de la imagen, se sabe que su anchura corresponde a 2 veces la altura debido a que el proceso de segmentación proporciona 2 imágenes cuadradas, se procede a pasar la escala de píxeles a una escala de posición unitaria (Fig. 26), donde el punto se define de acuerdo a su posición en el sistema coordenado: (m,n).

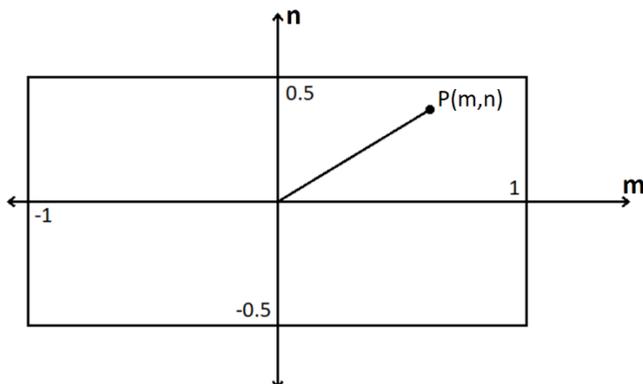


Figura 26. Posición de nueva coordenada P(m,n) en el plano equirrectangular. Fuente: Autor.

Estas magnitudes se traducen a latitud y longitud en la esfera abstracta y se hallan sus coordenadas en el punto en el que se encuentre el cursor en ese momento.

$$long = P_m * \pi \quad (6)$$

$$lat = P_n * \pi \quad (7)$$

$$P_x = \cos(lat) * \cos(long) \quad (8)$$

$$P_y = \cos(lat) * \sin(long) \quad (9)$$

$$P_z = \sin(lat) \quad (10)$$

Se devuelve esta coordenada a una relativa a la posición en el FishEye cambiando los ángulos (long,lat) a otros que se muevan con respecto al sistema coordenado de dicho plano: (θ, Φ) .

$$hipXZ = \sqrt{P_x^2 + P_z^2} \quad (11)$$

$$\theta = atan2(P_z, P_x) \quad (12)$$

$$\phi = atan2(hipXZ, P_y) \quad (13)$$

De esta manera se hallan las coordenadas en la escala unitaria en el FishEye y se encuentra la posición necesaria en términos del píxel en el que se encuentren. También se redondea dicha posición debido a que los pixeles se encuentran en una recta de números naturales y no puede haber números enteros.

$$r = 2 * \phi / \text{apertura} \quad (14)$$

$$x_1 = r * \cos(\theta) \quad (15)$$

$$y_1 = r * \sin(\theta) \quad (16)$$

La guía del producto indica que la apertura de las cámaras es de 190° , sin embargo, se encuentra experimentalmente que la apertura que realmente completa la información presente en la imagen es de 187° . La figura 27 ilustra el resultado de la aplicación de las ecuaciones de la proyección equirectangular, considerando este ángulo de apertura.



Figura 27. Resultado de proyección equirectangular. Fuente: Autor.

Como se mencionó previamente, la unión de las dos imágenes proyectadas genera una discontinuidad que se corrige con el proceso de cosido, para aplicarlo, se usa un par de

funciones de transferencia tipo rampa para modificar el brillo de ambas imágenes en la zona de unión en cada uno de sus 3 componentes de color. Estas funciones se describen de acuerdo a las ecuaciones **17** y **18**, respectivamente. Ambos valores de las funciones de transferencia representan líneas rectas que se mueven entre 0 y 1 (como se representa en la fig. 28b), para posteriormente mezclar los valores de color en la imagen final, como se ve en la ecuación **19**.

$$a_1 = \frac{\text{ancho} - x_{\text{actual}} + 1}{\text{ancho}} \quad (17)$$

$$a_2 = x_{\text{actual}} * \text{ancho} \quad (18)$$

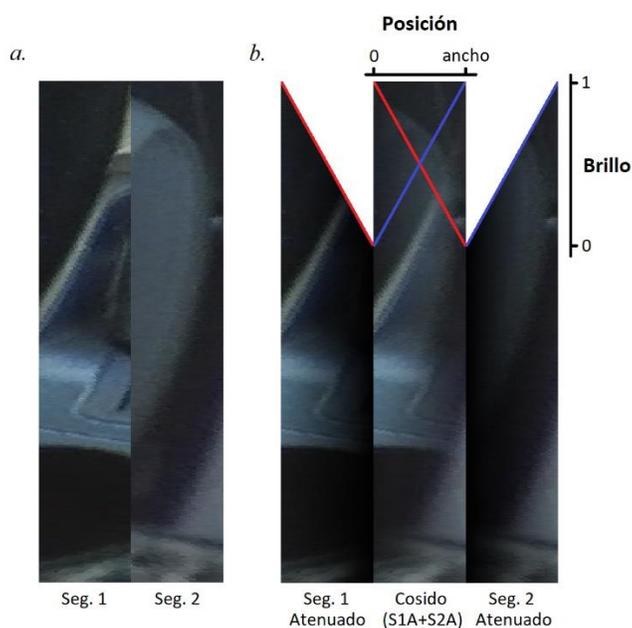


Figura 28. a. Segmentos de la imagen a fusionar. b. Atenuación y suma de segmentos en la zona del cosido. Fuente: Autor.

$$\text{zona} = a_1 * \text{zona}_1 + a_2 * \text{zona}_2 \quad (19)$$

Como resultado, la imagen producida (Fig. 29) es capaz de mezclar las áreas superpuestas correctamente y de acuerdo al campo de visión de la imagen utilizada, en esta

figura se presenta la imagen desplazada 90° hacia la izquierda en la esfera abstracta con el objetivo de apreciar el cosido en ambas zonas de unión, sin embargo, en el resultado final esto no es necesario.



Figura 29. Proyección equirectangular con bordes cosidos. Fuente: Autor.

Igualmente, la información obtenida permite ahora observar también la esfera pintando la superficie de la esfera de acuerdo a su latitud y altitud. Nótese que el cosido también ha sido aplicado en el borde entre mitades en la figura 30.



Figura 30. Esfera abstracta proyectada en un espacio tridimensional. Fuente: Autor.

3.2.1.2. Cálculo Hammer – Aitoff

Para calcular la proyección Hammer – Aitoff se debe usar la información presente en la esfera abstracta, es aquí donde se nota que precisamente la propiedad esférica de la proyección anteriormente calculada afirma que gracias a que es una traducción directa de sistemas coordenados, se pueden usar los valores en m y n de la anterior proyección como si fueran valores en latitud y longitud sin ningún problema.

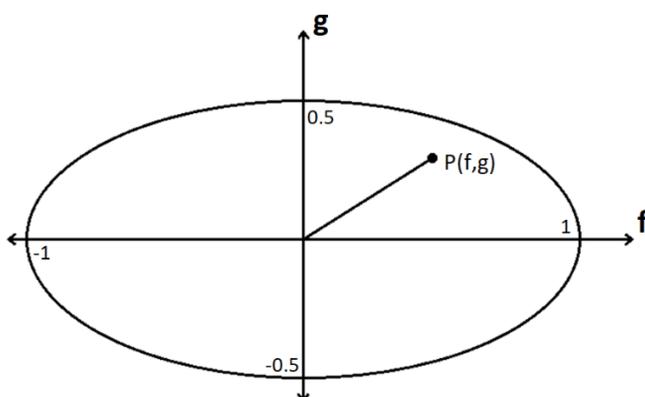


Figura 31. Posición de coordenada P(f,g) en el plano Hammer – Aitoff. Fuente: Autor.

Se reúsan las ecuaciones usadas para establecer los límites de la proyección equirectangular debido a que los límites de dicha imagen y la Hammer – Aitoff son los mismos (Fig. 31) y se halla la proyección por medio de la coordenada en h.

$$h_{mid} = \sqrt{1 - \frac{P_f^2}{2} - \frac{P_g^2}{2}} \quad (20)$$

$$long = 2 * \tan^{-1} \left(\frac{\sqrt{2} * h_{mid} * P_f}{2 * h_{mid}^2 - 1} \right) \quad (21)$$

$$lat = \sin^{-1}(\sqrt{2} * h_{mid} * P_g) \quad (22)$$

El resultado de este proceso es una imagen que funciona perfectamente dentro de los límites bajo los cuales la ecuación está establecida (de forma ovalada), pero proyecta

también píxeles por fuera de los bordes, donde la ecuación no fue planteada para funcionar y entrega resultados no deseados (la información empieza a repetirse una y otra vez), este segmento tiene que ser manualmente eliminado de la imagen volviendo píxeles negros todo lo que esté fuera del límite de la figura por medio de la ecuación **23** (la cual es la ecuación de una elipse cuando se supera la igualdad que le define, por lo tanto, el borde), la comparativa entre la imagen sin recortar y la imagen recortada se puede ver en la Figura 32.

$$SI \frac{(cursor_x - sicir)^2}{sicir^2} + \frac{(cursor_y - sicir/2)^2}{(sicir/2)^2} > 1 \text{ ENTONCES } pixel = 0 \text{ (23)}$$

a.



b.

Figura 32. Resultado de proyección Hammer sin (a) y con el límite aplicado (b). Fuente: Autor.

El diagrama de flujo del código descrito anteriormente se presenta de forma resumida en la figura 33.

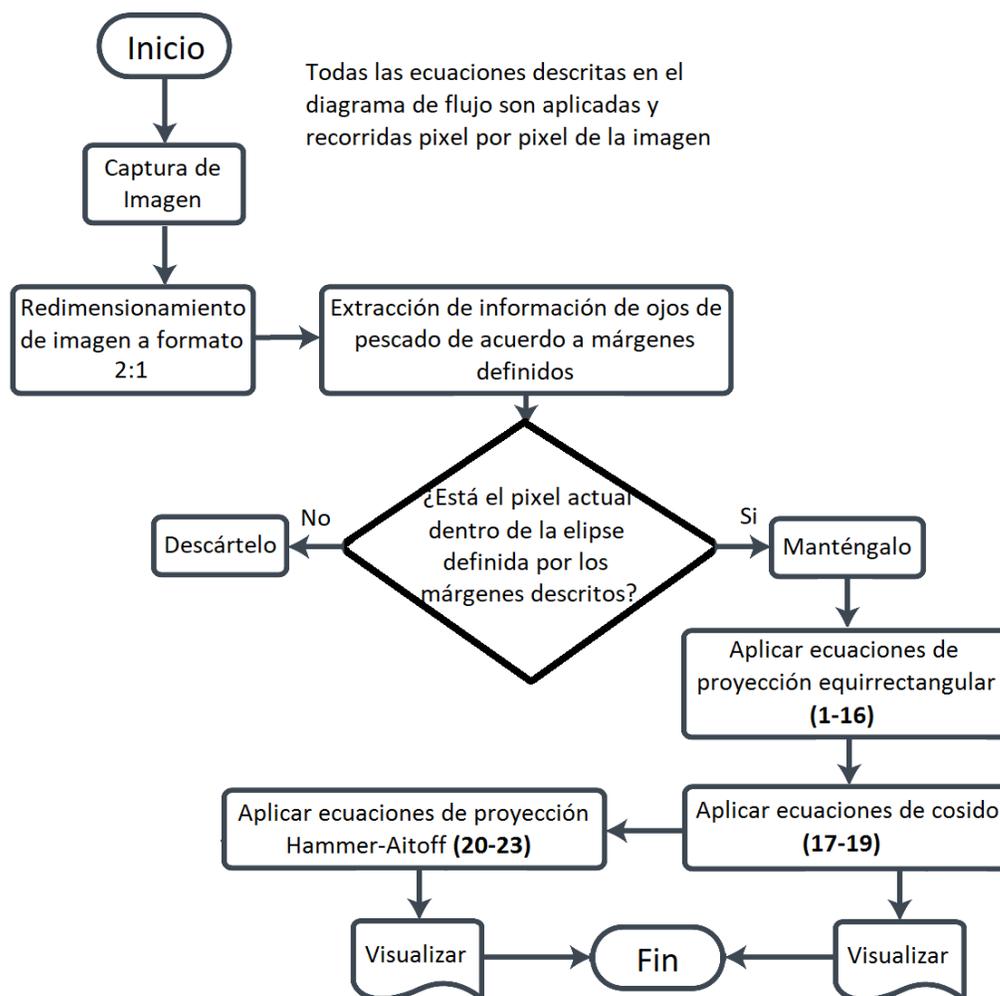


Figura 33. Diagrama de flujo del algoritmo calculado. Fuente: Autor.

3.3. Implementación en tiempo real

Las ecuaciones que permiten realizar las proyecciones descritas en las secciones anteriores fueron implementadas en el software MatLab, sin embargo, debido al tiempo de procesamiento requerido y a restricciones derivadas del uso de licencias, este no es una buena alternativa para una implementación en un ambiente real, por tal motivo, se hace la implementación en el lenguaje de programación JAVA.

A pesar de que la velocidad de procesamiento del código implementado en JAVA es 824 veces menor que en MatLab, tardando 17 ms por cuadro mientras que MatLab tarda 14 segundos, el hecho de que la CPU procese la información secuencialmente (por cada núcleo) sigue generando una limitante en la velocidad debido a el gran volumen de información que debe ser procesado en una ventana corta de tiempo para que las proyecciones puedan ser visualizadas en tiempo real, esto es, el tiempo tope que puede tardar una imagen para ser presentada y ser percibida a tiempo por el usuario de 50 ms (Trafton, 2014). El algoritmo usado en este proyecto para el cálculo de las proyecciones es de tipo repetitivo y necesita ser aplicado píxel por píxel, por esta razón se utiliza la tarjeta gráfica para aligerar la carga de trabajo del procesador.

La implementación de las ecuaciones descritas anteriormente se realiza usando OpenGL para que sea ejecutada sobre la GPU (Unidad de procesamiento gráfico) del computador. JAVA es empleado para crear y configurar la ventana de trabajo necesaria y recibir las imágenes de la cámara por medio de conexión IP, este proceso si es ejecutado sobre la CPU del computador.

La máquina empleada tiene una tarjeta gráfica con referencia *Gtx1050ti* de 4 Gb de VRAM, la cual usa arquitectura Pascal (Nvidia, 2017) y sigue teniendo soporte actualmente, posee también 768 núcleos CUDA (Kandrot & Sanders, 2010) que trabajarán simultáneamente para el procesamiento de las imágenes, lo que permite usar hasta la última versión de OpenGL (la 4.6, en el momento de escribir este documento), aunque se elige trabajar en la versión del preprocesador de shader #130, y por lo tanto se puede correr el

código escrito en este proyecto en tarjetas de video que soporten desde OpenGL 3.0 en adelante.

Esta implementación logra correr por encima de los 3000 cuadros por segundo si se permite al código usar todos los recursos que desee en la tarjeta de vídeo, sin embargo, si se espera a que la cámara usada entregue los cuadros a renderizar, es capaz de generar las imágenes al ritmo al que la cámara los genera (60 cuadros por segundo) y, por lo tanto, genera las proyecciones en vivo bajo una resolución 1280x720 a esa velocidad.

Es necesario tener en cuenta que la cantidad de píxeles (y la cantidad de procesamiento necesaria) aumenta con el cuadrado de la resolución de la imagen, y debido a que la resolución 4k (3840x2160) es 3 veces más grande en ambas direcciones que la de 720p (1280x720), renderizar en 4k a la misma tasa de cuadros, requiere usar 9 veces el poder de computación que fue necesario para renderizar en 720p. Por lo tanto, si renderizando a 720p se consigue una velocidad de 3000 cuadros por segundo, podemos suponer que a 4k, el computador podría llegar a los 350 cuadros por segundo si se usara todo el poder de procesamiento de la GPU.

4. Análisis de Resultados

En principio, el algoritmo fue implementado en MatLab, los valores que fueron necesarios para definir los bordes de la imagen durante su procesamiento fueron de radio en $X= 0.479$, radio en $Y= 0.484$, Offset en $X= 0.001$, distancia del centro de la imagen en $X= 0.5$ y distancia del centro de la imagen en $Y= -0.001$ para los bordes de la selección (Fig. 21), junto a 10° de separación para el cosido, experimentalmente se determinó que 187° grados de apertura funcionan mejor que los 190° que afirma la guía de la cámara usada; por otro lado se aprecia un pequeño desfase de un par de píxeles en el borde del círculo, lo cual indica que la figura es un elipsoide imperfecto, sin embargo tal desfase de tan solo píxeles resulta despreciable (como se puede ver en la fig. 34).



Figura 34. Desfase en borde de imagen con respecto a límite definido por código. Fuente: Autor.

Una vez calculadas se notan diferencias entre los resultados de ambas proyecciones, las cuales también varían de acuerdo al entorno capturado en la imagen y a la presencia del cosido. La proyección equirrectangular usa toda la imagen para proyectar los puntos de la

esfera abstracta y por lo tanto el nivel de definición es mucho mayor, sin embargo, el efecto de la deformación en los extremos se hace notar: objetos pequeños que se encuentran frente a la cámara (como su base y el tronco debajo de esta, para los ejemplos presentados en las Figuras 35 y 36) se ven amplificados enormemente.



Figura 35. Imagen 1 en proy. equirectangular con cosido y movida 90° a la izquierda. Fuente: Autor.



Figura 36. Imagen 2 en proy. equirectangular con cosido y movida 90° a la izquierda. Fuente: Autor.



Figura 37. Imagen 2 en proyección equirrectangular sin cosido. Fuente: Autor.

El uso de cosido causa diferencias notables en la zona de unión (como se puede apreciar al comparar las figuras 36 y 37), debido a que la diferencia entre la información percibida por ambas imágenes en dicha zona dependerá del campo de visión que se desea proyectar si no se usa cosido, mientras que el cosido usado en este proyecto permite amortiguar los efectos de dichas diferencias: en el caso de la imagen de ejemplo, hay un árbol que se duplica al no aplicar cosido debido a su distancia con respecto a la cámara.

De la misma manera, con la proyección Hammer, se pretende corregir este error, adjudicando menos zona libre a las zonas más pequeñas de la imagen (y por lo tanto reduciendo el área de la imagen que se utiliza), esto crea como efecto secundario una menor definición y nivel de detalle en la imagen en general (Fig. 38 y 39).



Figura 38. Imagen 1 en proyección Hammer con cosido y movida 90° a la izquierda. Fuente: Autor.



Figura 39. Imagen 2 en proyección Hammer con cosido y movida 90° a la izquierda. Fuente: Autor.

Otro defecto presente y que se puede apreciar con mucha mayor claridad en la figura 38 es la curvatura de zonas que de otra manera serían perfectamente rectas a medida que nos acercamos a los bordes izquierdo y derecho de la imagen, en este caso, el efecto deforma los edificios y arboles presentes en dichas zonas, los cuales son completamente rectos en la vida real.

El algoritmo fue implementado en partes separadas entre Java con el uso de OpenCV y GLSL (OpenGL Shading Language), este segundo conteniendo mayor parte del algoritmo y ejecutándose directamente dentro de la tarjeta de video, la cual trabaja independientemente, dedicándole a cada núcleo CUDA un pixel de la pantalla. Como resultado, se lograron cerca de 3000 cuadros por segundo en promedio sin sincronizar dicha tasa de refresco a la de la cámara usada, lo cual significa que la implementación en GLSL puede exportar con un retraso de tan solo 0.4 ms en el procesamiento de dicha imagen, en comparación con los 11 segundos necesarios en MatLab para producir una proyección equirrectangular y 14 segundos para una proyección Hammer.

La cámara exporta a 60 cuadros por segundo, es decir que cada imagen permanece en la pantalla durante $1/60$ seg o ~ 17 ms, teniendo en cuenta que el ser humano puede percibir la información recibida desde los 10 ms de exposición a la imagen y tiene una tolerancia media tope de 50 ms a el retraso (Trafton, 2014), el solo hecho de procesar las imágenes al ritmo en el que son recibidas es suficiente para garantizar responsividad en el usuario, sin embargo el retraso que hubo entre la captura de la imagen por la cámara y la muestra de la proyección en la pantalla fue en promedio de 2 segundos, lo cual constituye un retraso significativo para una proyección en tiempo real, este retraso se debe principalmente al preprocesamiento presente en la cámara antes de enviar la imagen y al hecho de que dichas imágenes se envían por conexión IP, ambos factores que están por fuera del alcance del algoritmo aplicado en este proyecto pero que necesitan ser corregidos a un tiempo tope de 30 ms si se desea lograr imágenes que se aproximen al tiempo real.

El enfoque usado en este proyecto permite manejar distintos modelos de sensores y proyecciones para implementar el algoritmo en ordenadores que permitan su ejecución, los resultados pueden descargarse en:
<https://mega.nz/#F!ohQ1xaLY!jSv4TMcgPblA2S1G96RLsw>

5. Conclusiones

En este proyecto se aplica un algoritmo de conversión de proyección de imagen con la finalidad de producir imágenes en 360° en tiempo real. Se implementaron 2 tipos de proyecciones (Equirrectangular y Hammer). La proyección cartográfica empleada, se constituye en un paso intermedio necesario para el tránsito entre proyecciones porque reduce la cantidad de puntos en los que se debe aplicar las ecuaciones de proyección.

La proyección equirrectangular mantiene las distancias en los ejes y las líneas rectas se mantienen como tal, sin embargo, la información se ve distorsionada y amplificada a medida que se aleja de la línea de latitud 0, haciendo que se pierdan las relaciones de tamaño presentes en la imagen. Por otro lado, la proyección Hammer no es capaz de mantener los ángulos de la imagen original y por tanto las líneas rectas se curvan y convergen en los polos de la esfera de proyección, pero preserva las proporciones, logrando así que las áreas de todas las figuras se mantengan, haciendo que este tipo de proyección sea útil en aplicaciones de vigilancia y reconocimiento del entorno.

A pesar de que los algoritmos de proyección pueden implementarse para su uso en tiempo real, existen limitantes externas como el proceso que realiza el firmware de la cámara que retrasan la imagen procesada con respecto a la imagen de origen.

El cosido usado en el proyecto no es de naturaleza adaptativa ni permite corregir los elementos en la pantalla de acuerdo a la distancia así como no puede ser aplicado en vivo, esto es debido a que dichas implementaciones usan procesos iterativos, lo cual no permite o limita la proyección de las imágenes en tiempo real, por tanto podría considerarse el uso

de otros métodos de detección de objetos en tiempo real que aprovechen la naturaleza continua del formato para intentar corregir e optimizar el cosido presente en la imagen.

El algoritmo aplicado en este proyecto se proyecta como un paso hacia adelante necesario para lograr cumplir el objetivo de que sea una herramienta útil para complementar el ángulo de visión disponible en la visión humana debido a su aplicación de conceptos abstractos y altamente generalizables a el proceso de proyección que se necesite.

En este sentido, mientras que todas las referencias mostradas en los antecedentes (ver sección 2) son relevantes a esta implementación, quizás las más relevantes son aquellas que implementan proyecciones en 360° en armazones diseñados para la cabeza humana, como es el caso del LiveSphere (Kasahara, Nagai , & Rekimoto, 2014), los cuales ya logran capturar la información proveniente de los alrededores del usuario pero lo hacen con distintos objetivos, que van desde la grabación hasta la posterior interacción, y por lo tanto, este es el primer proyecto que tiene el objetivo de aumentar el campo de visión para su uso por el usuario en tiempo real.

6. Recomendaciones

Se recomienda optimizar la implementación del código con la finalidad de permitir una ejecución en vivo en sistemas embebidos o de naturaleza portable, y permitir la implementación en dispositivos de visualización de realidad virtual, donde una mayor resolución de renderizado y una más alta tasa de actualización son necesarios.

También se recomienda hacer un estudio estadístico con respecto a la comodidad y la respuesta de seres humanos con respecto al uso de sistemas de realidad virtual con proyecciones en 360° de acuerdo a distintos tiempos de exposición a las imágenes y a la proyección que se presente al usuario.

7. Bibliografía

- Banks, M., Sprague, W., Schmoll, J., Parnell, J., & Love, G. (2015, Agosto 7). Why do animal eyes have pupils of different shapes? *Science Advances*.
- Belbachir, Schraml, Mayerhofer, & Hofstatter. (2018). *A Novel HDR Depth Camera for Real-Time 3D 360° Panoramic Vision*.
- Bourke, P. (2016, Agosto). *Paul Bourke*. Retrieved from <http://paulbourke.net/dome/dualfish2sphere/>
- Brand , R., Diniz, B., Ribeiro, R., & Humayun, M. (2012). *Artificial vision through neuronal stimulation*.
- Chessa, M., Noceti, N., Odone , F., Solari, F., Sosa, J., & Zini, L. (2015). *An integrated artificial vision framework for assisting visually impaired users*.
- De Francisco, O., Sánchez, H., Estrems, M., & Carrero-Blanco, J. (2017). *Position precision improvement throughout controlled led paths by artificial vision in micromachining processes*. .
- Deligiannidis, L., & Arabnia, H. (2014). *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*. Morgan Kaufmann.
- Deligiannidis, L., & Arabnia, H. (2014). *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*. Morgan Kaufmann.
- Dobelle, W. (2000). *Artificial Vision for the Blind by Connecting a Television Camera to the Visual Cortex*. *ASAIJ Journal: January-February 2000 - Volume 46 - Issue 1 - p 3-9*. *ASAIJ Journal: - Volume 46 - Issue 1 - p 3-9*.
- Dobelle, W. H. (2000).

García, F., & Morales, A. (2016). *Towards an artificial vision-robotic system for tomato identification.*

Gómez, J., González, D., Gonzalo, A., Herrero, J., Muñoz, Á., & del Val, J. (2010). *Reconstrucción 3D y realidad virtual en criminología.*

Goodale, M., & Westwood, D. (2004, Abril). An evolving view of duplex vision: separate but interacting cortical pathways for perception and action. *Current Opinion in Neurobiology*, 14(2), 203-211.

Grafarend, & Syffus. (1997). The Hammer projection of the ellipsoid of revolution. *Journal of Geodesy.*

Heinrichs, Montello, Smith, Sullivan, Thompson, & Pick. (1995). *Topographic map reading.*

Heras lara, L., & Villareal Benítez, J. (2004). La realidad aumentada: una tecnología en espera de usuarios. *Revista Digital Universitaria*, vol. 5, núm. 7, .

Ho, T., & Budagavi, M. (2017). Dual-fisheye Lens Stitching for 360-degree Imaging. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New Orleans, LA, USA.

Hughes, J., Van Dam, A., Mcguire, M., Sklar, D., Foley, J., Feiner, S., & Akeley, K. (2014). *Computer Graphics* .

Hunt, C. (2018, Julio 25). Field of view face-off: Rift vs Vive vs Gear VR vs PSVR. *VrHeads.*

Infaimon. (2018). *Soluciones de visión artificial.*

Ito. (2010). *a 360-Degree Autostereoscopic Display Prototype.*

- Kandrot, E., & Sanders, J. (2010). *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley Professional.
- Kasahara, S., Nagai, S., & Rekimoto, J. (2014). *LiveSphere: immersive experience sharing with 360 degrees head-mounted cameras*.
- Levin, A., Zomet, A., Peleg, S., & Weiss, Y. (2004). Seamless Image Stitching in the Gradient Domain. *Computer Vision - ECCV 2004*, 377-389.
- Lhuillier, M., & Nguyen, T.-T. (2015). *Synchronization and Self-Calibration for Helmet-Held Consumer Cameras, Applications to Immersive 3D Modeling and 360 Video*.
- Likert, R. (1932). A technique for the measurement of attitudes. *Archives of psychology*.
- Loaiza Quintana, A. F., Manzano Herrera, D. A., & Múnera Salazar, L. E. (2012, Septiembre-Diciembre). Sistema de visión artificial para conteo de objetos en movimiento. *El Hombre y la Máquina*(40), 87-101.
- Londoño-Osorio, V., Marín-Pineda, J., & Arango-Zuluaga, E. (2013, Octubre). Introducción a la Visión Artificial mediante Prácticas de Laboratorio Diseñadas en Matlab. *Tecno Lógicas*, 591-603.
- Manis, T., & Choi, D. (2018). *The virtual reality hardware acceptance model (VR-HAM): Extending and individuating the technology acceptance model (TAM) for virtual reality hardware*.
- Nagata, S. (1996, Diciembre). The binocular fusion of human vision on stereoscopic displays - Field of view and environment effects. *Ergonomics*, 39(11), 1273-1284.

- Napieralla, J. (2018). *Comparing Graphical Projection Methods at High Degrees of Field of View*. Tesis, Blekinge Institute of Technology, Faculty of Computing, Karlskrona. Retrieved from <http://www.diva-portal.se/smash/get/diva2:1229190/FULLTEXT02.pdf>
- Nvidia. (2017). Ultra-Performance Pascal GPU and NVLink Interconnect. *IEEE*, 7-17.
- Orozco, C., Hernandez, E., & Trefftz. (2006). *Collaborative and distributed augmented reality in teaching multi-variate calculus*. ACTA Press Anaheim, CA, USA, : WBE'06 Proceedings of the 5th IASTED international conference on Webbased education.
- Ortigoza-Ayala, L. O., Ruiz-Huerta, L., & Caballero-Ruiz, A. (2009). Visión artificial en el humano. *Revista Médica del Instituto Mexicano del Seguro Social*, 47(4), 393-398.
- Palmer, A. (2009, Noviembre 17). Negotiation and Resistance in Global Networks: The 1884 International Meridian Conference. pp. 7-24.
- Peng, X., Bennamoun, M., Wang, Q., Ma, Q., & Zhiyong, X. (2014). *A Low-Cost Implementation of a 360° Vision Distributed Aperture System*.
- Racette, L., Fischer, M., Bebie, H., Holló, G., Johnson, C., & Matsumoto, C. (2016). *Visual field digest: A guide to perimetry and the Octopus perimeter*. Köniz, Suecia: Haag-Streit AG. Retrieved from https://www.haag-streit.com/fileadmin/Haag-Streit_USA/Visual_Field_Digest_6th_edition_2017.pdf

Snyder, & Voxland. (1989). *An album of map projections*. US Government Printing Office.

Snyder, J., & Bugayevskiy, L. (1995). *Map Projections - A Reference Manual*.

Tomás, R., Ferreiro, J., Sentana, I., & Díaz, M. (2002). *Aplicaciones de la proyección estereográfica en ingeniería geológica*.

Trafton, A. (2014, Enero 16). In the blink of an eye. *MIT News Office*.

Yagi, A., Imura, M., Kuroda, Y., & Oshiro, O. (2011). *360-degree fog projection interactive display*.

Zhu, Y., Zhai, G., & Min, X. (2018). *The prediction of head and eye movements for 360 degree images*.