

**Información general**

Facultad Ciencias Naturales e Ingenierías			
Programa académico Ingeniería Electrónica		Grupo(s) de investigación GICAV	
Nombre del semillero / Siglas ROBUTS	Fecha creación: 2013	<p>Logo</p>	
	Campus:		
Líneas de Investigación Automatización y Sistemas de Control Avanzados Robótica			
Áreas del saber (1)			
<input type="checkbox"/>	1. Ciencias Naturales	<input checked="" type="checkbox"/>	2. Ingeniería y Tecnologías
<input type="checkbox"/>	3. Ciencias médicas y de la salud	<input type="checkbox"/>	4. Ciencias Agrícolas
<input type="checkbox"/>	5. Ciencias sociales	<input type="checkbox"/>	6. Humanidades

**Información del director del proyecto**

Nombre Carlos Iván Nova Pinzón		No. de identificación y lugar de expedición 1100954253 San gil	
Nivel de formación académica Ingeniero Electrónico		<input checked="" type="checkbox"/>	Asesor
		<input checked="" type="checkbox"/>	Líder de semillero
Celular 3142597807		Correo Electrónico cnova@correo.uts.edu.co	

**Información de los autores**

Nombre y/o firma	No. Identificación y lugar de expedición:	Celular	Correo electrónico
Jefferson Mora Meza	1095811209	3157785416	jeffersonmora@outlook.com

**Proyecto**

1. Título del proyecto	Modalidad del Proyecto (2)
------------------------	----------------------------

DISEÑO DE INTERFAZ GRAFICA PARA EL PROCESAMIENTO DE IMÁGENES EN 3D

PA	PI	TG	RE	Otra. ¿Cuál?
	x			

2. Planteamiento de la Problemática: Los laboratorios son espacios fundamentales para el desarrollo practico de las actividades institucionales de los estudiantes cursantes, así como los grupos de investigación de las Unidades Tecnológicas de Santander en adelante UTS.

Para esto y con base en el fortalecimiento institucional las UTS ha adquirido nuevas herramientas que permitan abordar nuevos campos de desarrollo, entre estas se encuentra un equipo body scan 3D, del cual surge la necesidad de intervenir.

La problemática se basa en que este equipo cuenta con un software muy limitado, lo que impacta en las actividades que se desean realizar en la institución. Adicionalmente la licencia de un (1) año, afecta económicamente (\$499 USD mensuales) el poder usar esta herramienta de forma constante.

3. Antecedentes: Reconstrucción 3D

La visión computarizada es aplicada a múltiples campos en la actualidad, la medicina y la industria automovilística son un claro ejemplo del interés por estos campos en particular. La reconstrucción 3D es el proceso en el cual se busca dotar de características físicas como la forma, volumen, profundidad y apariencia a una imagen presentada en una pantalla (Palos, 2012). Para esto se requieren de varias técnicas y/o métodos de mallado 3D.

El proceso de reconstrucción puede ser llevado a cabo mediante métodos activos o pasivos. En estos métodos es necesario un mecanismo de digitalización del objeto 3D real. Un punto interesante de estos métodos es la digitalización del objeto 3D a partir de una imagen 2D del mismo (Camacho Rojas, 2013).

Los métodos activos que en resumen están en contacto con el objeto de que se quiere reproducir, por ejemplo, sensores de ultrasonido, señores laser, o incluso sensores mecánicos son utilizados en ampliamente en la industria de los hidrocarburos para encontrar defectos al interior de tuberías de transporte. Los scanner o equipos utilizados en los trabajos de integridad mecánica para la detección de fallas en las tuberías de transporte de hidrocarburos pueden utilizar sensores laser, de ultrasonido y de campo magnético. Ha esto se le denomina métodos activos porque están en contacto con el objeto reconstruido. Los métodos pasivos por el contrario no interfieren con el objeto de que se desea reconstruir para este caso se usan una o más imágenes digitales, para reconstruir y obtener un modelo 3D.

Para la reconstrucción 3D a partir de imágenes 2D se hace necesario hacer uso de múltiples vistas en las cuales se elige puntos de referencia o de interés y por medio de triangulación se puede proceder a su reconstrucción, proceso que a su vez depende de la posición y ajuste de la cámara utilizada para tal fin (Wong, s.f.).

El primer paso para realizar una reconstrucción tridimensional de una escena a partir de imágenes de la misma es encontrar correspondencias entre dichas imágenes. Para obtener dichas correspondencias es necesario detectar puntos de interés, estos son, informalmente, puntos seleccionados sobre zonas notables de la imagen, como, por ejemplo, esquinas. Dichos puntos, además, deben ser robustos a diversas perturbaciones, por ejemplo, la escala o las rotaciones (García, A, 2019, p11).

De esto nacen sistemas que van desde lo más básico a nivel aficionado con cámaras de celular, webcams hasta un nivel profesional como son las estructura para la captura de imágenes fotogramétricas con cámaras DSLR.

En el ámbito nacional se han desarrollado diversos trabajos en múltiples campos de estudios donde se aplican técnicas de reconstrucción tridimensional. Es el caso de la Universidad del Norte donde se deseaba construir un volumen tridimensional del corazón a partir de imágenes TAC y seleccionar a la aorta como una de sus regiones. En este se utilizaron diferentes algoritmos para procesar las imágenes adquiridas en formato DICOM. Proceso llevado a cabo por medio del software PYTHON en el cual se buscaron librerías asociadas al procesamiento de imágenes. Asimismo, se hizo necesario diseñar y construir una interfaz gráfica en la misma plataforma para visualiza los resultados obtenidos (Mercado, L. 2018).

Los dispositivos móviles que se comercializan actualmente han incorporado un sin número de técnicas de procesamiento de imágenes para mejorar la experiencia al usuario. Entre estas encontramos filtros, correctores de escena e incluso secuencias tridimensionales haciendo uso de las múltiples caramas que permiten generar imágenes estereoscópicas.

(Revelo Luna, D; Usama, F; Flórez Marulanda, J. 2012) trabajaron en técnicas de basadas en estereoscopía para la reconstrucción tridimensional de imágenes reales y en técnicas de disparidad por extracción de características para la representación de puntos en un escenario 3D virtual.

Interfaz gráfica de usuario – GUI

La interfaz gráfica de un software, juega un papel importante en la interacción que tiene el usuario con las diferentes herramientas y aplicativos del software. La interfaz gráfica de usuario, conocida también como GUI, es un programa informático que utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador (Grayson, 2000).

A nivel local, nacional e internacional se han desarrollado una gran cantidad de aplicativos GUI, un claro ejemplo son la GUI diseñadas en la asignatura de robótica o procesamiento digital de imágenes por los estudiantes para el desarrollo de la asignatura. Actualmente existen una gran cantidad de lenguajes de programación que se pueden utilizar para el desarrollo de interfaces gráficas, pero uno que sea robusto, de fácil interpretación y libre es Python.

Python es un lenguaje de programación de propósito general muy poderoso y flexible, a la vez que sencillo y fácil de aprender. Es un lenguaje de alto nivel, que permite procesar fácilmente todo tipo de estructuras de datos, tanto numéricos como de texto. Python se desarrolla bajo una licencia de open source o código abierto aprobada por OSI, por lo que se puede usar y distribuir libremente, incluso para uso comercial. Tanto la biblioteca estándar de Python como los módulos aportados por la comunidad permiten infinitas posibilidades (Covantec, 2018).

- Desarrollo web e Internet.
- Acceso a la base de datos.
- GUI's de escritorio.
- Científico y numérico.
- Educación.
- Programación de red.
- Desarrollo de Software y Juegos.

El presente trabajo en las UTS propone el diseño de una interfaz gráfica para el procesamiento de imágenes en 3D, que va desde la captura de imágenes desde un sistema body scan que cuanta con módulos de cámaras Raspberry hasta la renderización de un objeto en pantalla haciendo uso de plataformas de desarrollo de código abierto.

4. Justificación: Como se expone en el estado del arte el procesamiento digital de imágenes ha impactado notablemente en la ciencia, como en la industria en general. Actualmente esta área del conocimiento ha crecido al punto que impacta en las diferentes actividades del día a día. Es por eso que la institución ha venido promoviendo el enfoque de estos campos a los programas que actualmente imparte con la adquisición de nuevo material tecnológico.

Para dar solución a la problemática que actualmente se presenta se propone desarrollar una interfaz gráfica basada en software libre que permita realizar el procesamiento y reconstrucción de imágenes en 3D a partir de múltiples imágenes 2D. Esto permitirá inicialmente reducir los costos de licenciamiento mensual, ya que estará desarrollado en una plataforma de software libre.

El desarrollar en plataforma libre permitirá que se pueda enfocar las características de la plataforma como el diseño, la interacción e incluso las herramientas a disposición del usuario. Asimismo, la aplicación a diseñar permitirá que otras áreas del conocimiento puedan hacer uso de este equipo, eliminando la limitación del software actual.

5. Marcos Referenciales:

FOTOGRAMETRÍA DIGITAL

Etimológicamente la palabra fotogrametría se deriva de tres raíces griegas: foto, luz y grama, que significan lo que se ha dibujado, y metría, como medir (Sifuentes R. & Vásquez A., 1997) (Cheli, 2021) (Soler Alemán, 2013). Es decir, la fotogrametría significa medir gráficamente por medio de la luz (Soler Alemán, 2013). En otras palabras, la técnica busca extraer la mayor cantidad de información sobre la posición de un objeto en una fotografía.

Según la Sociedad Americana de Fotogrametría y Sensores Remotos, ASPRS por sus siglas en inglés, la fotogrametría es el arte, la ciencia y la tecnología de obtener información fidedigna de objetos físicos y su entorno mediante procedimientos de grabación, medida e interpretación de imágenes fotográficas (Arévalo Vera, Bayona Ibáñez, & Rincón Parada, 2015).

El principio general de la fotogrametría se basa en la visión estereoscópica. Según (Cheli, 2021), este tipo de visión es una facultad de los seres humanos que le permite ver relieve en los objetos que observa con sus ojos. Mientras que afirma que, “debido a que los humanos tienen dos ojos y las pupilas están separada unos 6,3 centímetros, el mundo se ve de forma estereoscópica, es decir, cada ojo ve un objeto ligeramente diferente. Para formar una sola imagen en 3D que transmita profundidad, volumen, distancia, altura y anchura, el cerebro fusiona la imagen registrada por un ojo con la imagen registrada por el otro”

El principio de la visión estereoscópica se aplica en fotogrametría con el uso de pares fotogramétricos, es decir, haciendo uso de dos imágenes tomadas al mismo objeto, separadas por cierta distancia entre sí.

La fotogrametría digital se basa en la utilización de imágenes digitales. Una imagen digital es simplemente un mapa de píxeles. Y un mapa de píxeles es una matriz donde cada punto está ordenado en el espacio y codificado unitariamente (Medina Carrillo, 2014). Es así que la fotogrametría digital convierte las imágenes bidimensionales en modelos tridimensionales permitiendo observar contornos, texturas y volúmenes de la escena capturada (Roldán L., 2019).

#### OPENMVG

OpenMVG es una biblioteca para científicos de visión por computadora dirigida a la comunidad de geometría de múltiples vistas. Está diseñado para proporcionar un fácil acceso a:

- Solucionadores precisos de problemas de geometría de múltiples vistas.
- Pequeñas bibliotecas para realizar tareas desde la detección / coincidencia de características hasta la estructura de movimiento.
- Estructura completa de las tuberías de movimiento.

OpenMVG proporciona una colección de pequeñas bibliotecas que permiten resolver problemas de visión por computadora y construir una tubería SM completa (Moulon, Monasse, Marlet, & Otros, s.f.).

#### 2.1.3. OPENMVS

Si bien hay proyectos maduros y completos de código abierto dirigidos a tuberías de estructura a partir de movimiento (como OpenMVG), no hay ninguno que aborde la última parte de la fotogrametría. OpenMVS tiene como objetivo llenar ese vacío proporcionando un conjunto completo de algoritmos para recuperar la superficie completa de la escena a reconstruir. La entrada es un conjunto de poses de cámara más la nube de puntos dispersa y la salida es una malla texturizada. Los principales temas cubiertos por esta biblioteca son (Cernea, 2020):

- Reconstrucción densa de nubes de puntos para obtener una nube de puntos lo más completa y precisa posible.
- Reconstrucción de malla para estimar una superficie de malla que explique mejor la nube de puntos de entrada.
- Refinamiento de la malla para recuperar todos los detalles finos.
- Texturizado de malla para calcular una textura nítida y precisa para colorear la malla.

#### CÁMARA PINHOLE Y PARAMETROS INTRINSECOS

La mayoría de cámaras fotografías describen relativamente bien el modelo ideal de cámara pinhole o estenopeica. El proceso geométrico de formación de imágenes en la cámara pinhole es determinado eligiendo un centro de proyección en perspectiva y un plano retiniano. La proyección de un punto de la escena se obtiene como la intersección de una línea que pasa por este punto y el centro de proyección  $C$  con el plano retiniano  $R$  (Pollefeys, 2004).

En el caso más simple en el que el centro de proyección se coloca en el origen del marco del mundo y el plano de la imagen es el plano  $Z=1$ , el proceso de proyección se puede modelar de la siguiente manera:

$$x=XZ \quad y=YZ$$

#### TRANSFORMACIÓN DE CARACTERÍSTICAS INVARIANTES DE ESCALA - SIFT

La coincidencia de imágenes es un aspecto fundamental en la visión por computadora. El método SIFT permite extraer las características más relevantes de imágenes con las cuales se puedan relacionar diferentes vistas de un objeto de forma confiable. Las características que se hallan por este método son invariantes a la escala y a la rotación, y parcialmente son invariantes a la iluminación y al punto de vista de cámara 3D (Lowe, 2004).

A continuación, se presentan las cuatro principales etapas para llevar a cabo la búsqueda o el cálculo del conjunto de características para cualquier imagen.

#### DETECCIÓN DE EXTREMOS DE ESPACIO DE ESCALA

La primera etapa de la detección de puntos clave busca identificar las ubicaciones y escalas que se pueden asignar repetidamente bajo diferentes vistas del mismo objeto. Koenderink (1984) y Lindeberg (1994) demostraron que, bajo una variedad de supuestos razonables, el único núcleo de espacio de escala posible es la función gaussiana.

#### DESCRIPTOR DE PUNTOS CLAVE

Las operaciones anteriores han asignado una ubicación, escala y orientación de la imagen a cada punto clave. Estos parámetros imponen un sistema de coordenadas 2D local repetible en el que describir la región de la imagen local y, por lo tanto, proporcionan invariancia a estos parámetros. Los gradientes de la imagen local se miden en la escala seleccionada en la región alrededor de cada punto clave. Estos se transforman en una representación que permite niveles significativos de distorsión de forma local y cambio en la iluminación (Lowe, 2004).

#### ACCELERATED KAZE – AKAZE

Los enfoques de descripción y detección de características multi-escala más conocidos son SIFT y SURF, siendo SURF menos exigente computacionalmente que SIFT. Ambos enfoques hacen uso del espacio de escala gaussiana, ya sea construyendo el espacio de escala gaussiana en un marco piramidal como en SIFT, o aproximando derivadas

gaussianas a través de filtros de caja como en SURF. El principal inconveniente se debe a que el desenfoque gaussiano no conserva los límites del objeto y suaviza en la misma medida los detalles y el ruido en todas las escalas, lo que reduce la precisión y el carácter distintivo de la localización (Alcantarilla, Nuevo, & Bartolí, 2013).

Para superar este problema, se han presentado algunos métodos que tienen como objetivo detectar y describir características en espacios de escala no lineales. Haciendo que el desenfoque se adapte localmente a los datos de la imagen, desenfocando pequeños detalles, pero conservando los límites del objeto como lo hace KAZE, el principal inconveniente de KAZE es que es computacionalmente pesado. Basados en obtener características con baja demanda computacional haciendo uso del filtrado no lineal, es que Alcantarilla & et al. (2013) presentan Accelerated KAZE (AKAZE), el cual se soporta en un marco matemático llamado Difusión Explícita Rápida (FED por sus siglas en inglés) integrado en un marco piramidal (Alcantarilla, Nuevo, & Bartolí, 2013).

#### a. Filtrado de difusión no lineal:

El filtrado de difusión no lineal describe la evolución de la luminancia de una imagen a través del aumento de los niveles de escala como la divergencia de una determinada función de flujo que controla el proceso de difusión. Estos enfoques se describen mediante ecuaciones diferenciales parciales no lineales (PDE), debido a la naturaleza no lineal de las ecuaciones diferenciales involucradas que difunden la luminancia de la imagen a través del espacio de escala no lineal

### CONSTRUCCIÓN DE UN ESPACIO DE ESCALA NO LINEAL CON DIFUSIÓN EXPLÍCITA RÁPIDA

En primer lugar, se debe definir un conjunto de tiempos de evolución a partir de los cuales se pueda construir el espacio de escala no lineal. El espacio de escala está discretizado en una serie de octavas  $O$  y subniveles  $S$ . El conjunto de octavas y subniveles se identifican mediante un índice de octava discreto  $o$  y un subnivel uno  $s$ .

### DESCRIPCIÓN DE CARACTERÍSTICAS

Los descriptores binarios (como los que se utilizan en BRIEF, ORB y BRISK) han disfrutado de un uso generalizado últimamente, ya que pueden calcularse y combinarse de manera muy eficiente. A pesar de esto, solo recientemente se ha obtenido información sobre cómo funcionan realmente estas características binarias. De esta manera Alcantarilla & et al. (2013) utilizan el binario de diferencia local (MLDB) que extrae la información de gradiente e intensidad del espacio de escala no lineal. En este además de los valores de intensidad, se utilizan la media de las derivadas horizontal y vertical en las áreas que se comparan, lo que da como resultado 3 bits por comparación (Alcantarilla, Nuevo, & Bartolí, 2013).

LDB propone que se utilicen varias cuadrículas de pasos más finos, dividiendo el parche en cuadrículas de  $2 \times 2$ ,  $3 \times 3$ ,  $4 \times 4$ , etc., como se muestra en la Figura 13(a). Los promedios de esas subdivisiones son muy rápidos de calcular utilizando imágenes integrales si el descriptor está en posición vertical (no invariante en rotación). Sin embargo, cuando se considera la rotación de los puntos clave, no se pueden utilizar imágenes integrales, y visitar todos los puntos en una subdivisión rotada puede resultar relativamente costoso en tiempo de cálculo. En lugar de usar el promedio de todos los píxeles dentro de cada subdivisión de la cuadrícula, las cuadrículas son sub-muestreadas en pasos que son una función de la escala  $\sigma$  de la característica. El muestreo dependiente de la escala, a su vez, hace que el descriptor sea robusto a los cambios de escala. Este proceso se describe en la Figura 13(b). M-LDB utiliza las derivadas calculadas en el paso de detección.

### GEOMETRÍA DE MÚLTIPLE VISTA

Dadas diferentes vistas de una escena se puede observar que existen varias relaciones entre dos, tres o más imágenes. Esto es muy importante para la calibración y reconstrucción 3D a partir de imágenes (Pollefeys, 2004).

### DOS VISTAS GEOMÉTRICAS

Dado un punto de imagen en una imagen, ¿esto restringe la posición del punto de imagen correspondiente en otra imagen? Resulta que sí y esta relación se puede obtener de la calibración o incluso de un conjunto de características de cada imagen.

Aunque la posición exacta del punto de la escena  $M$  es desconocido, es probable que esté en la línea de visión del punto de imagen correspondiente  $m$ . Esta línea se puede proyectar sobre otra imagen y el punto correspondiente  $m'$  está obligado a estar en esta línea proyectada  $l'$  esto se ilustra en la Figura 8. De hecho, todos los puntos del plano  $\pi$  definidos por los dos centros de proyección y  $M$  tienen su imagen

### 6. Objetivo General y Objetivos específicos:

Diseñar una interfaz gráfica para el procesamiento de imágenes en 3D con software libre  
Objetivos específicos

- Identificar el software libre adecuado para el procesamiento de imágenes.
- Diseñar interfaz gráfica que permita reconstruir imágenes en 3D
- Validar el proceso de reconstrucción a con el software de AUTODESK

**7. Metodología:**

Para el diseño del presente trabajo de grado, se parte desde los medios disponibles en las Unidades Tecnológicas de Santander, siendo este un escáner de cuerpo completo móvil Twinstant que en resumen es un conjunto de cámaras distribuidas uniformemente en un área circunferencial el cual permite capturar imágenes simultáneamente dentro de su área de trabajo.

En función del conjunto imágenes entregadas por esta herramienta, se inicia un proceso exploratorio de los diferentes métodos que existen actualmente para la reconstrucción de modelos tridimensionales, con el fin de seleccionar el más adecuado y eficiente para realizar dicho proceso. Una vez seleccionado se toma un enfoque mixto con el fin de verificar la efectividad del proceso de reconstrucción tridimensional a partir de los registros obtenidos por el método usado. La técnica se basa en experimentar con las diferentes opciones disponibles de las librerías seleccionadas y los diferentes conjuntos de datos de entrada para un posterior análisis visual de los resultados obtenidos. En la figura 20 se presentan las etapas llevadas a cabo para el desarrollo del presente trabajo de grado.

Seguido a esto para el desarrollo del aplicativo de escritorio se selecciona un lenguaje de programación de fácil comprensión, suficientemente robusto y eficiente para integrar la interacción del usuario con cada una de las opciones y métodos disponibles por la librería y dependencias.

**8. Avances realizados:**

Técnicas de censado y reconstrucción 3D a partir de software libre

**9. Resultados esperados:** Desarrollo de interfaz gráfica que permite la captura de imágenes a partir del equipo Body Scan 3D y reconstruir el objeto, teniendo en cuenta técnicas de reconstrucción con software libre.

**10. Cronograma:**

Actividad (Semanal)	Fase 1				Fase 2					Fase 3					Fase 4				Fase 5			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Revisión bibliográfica	█																					
Seleccionar software		█	█	█																		
Diseño de interfaz gráfica					█	█	█	█	█													
Validación de software										█	█	█	█	█	█	█						
Entrega del documento Final para evaluación																	█	█				
Sustentación del trabajo de grado																			█	█		
Entrega final																				█	█	█

**12. Bibliografía:**

Grandón Pastén, N., Aracena Pizarro, D., & Tozzi, C. L. (2007). Reconstrucción de objeto 3D a partir de imágenes calibradas. Artículo, Revista Chilena de Ingeniería, Arica. Obtenido de <http://dx.doi.org/10.4067/S0718-33052007000200006>

Bescós Torcal, B. (2015). Reconstrucción de escenas 3D a partir de imágenes. Trabajo fin de grado, Universidad de Zaragoza. Obtenido de <http://zaguan.unizar.es/record/47529?ln=es#>

Camacho Rojas, L. F. (2013). Sistema de reconstrucción 3D multicámara. Tesis, Centro de investigaciones en óptica AC. Obtenido de <https://bibliotecas.cio.mx/buscar.php>

Castillo Zaragoza, J. M. (2019). Sistema de reconstrucción 3D del cuerpo humano a partir de múltiples vistas RGB-D. Trabajo de fin de grado, Universidad de Alicante, Alicante. Obtenido de <http://rua.ua.es/dspace/handle/10045/93559>

Cerón Correa, A. (2017). Desarrollo de un sistema de visión monocular para UAVs: aplicación al seguimiento de líneas eléctricas y reconstrucción del terreno asociado. Tesis, Universidad Nacional de Colombia, Bogotá. Obtenido de <http://bdigital.unal.edu.co/55414/>

- Covantec. (2018). Programación en Python - Nivel básico. Obtenido de <https://entrenamiento-python-basico.readthedocs.io/es/latest/>
- Dapoto, S., Cristina, F., Vegas, J., Artola, V., Russo, C., & Abásolo, M. J. (s.f.). Reconstrucción 3D: de la adquisición a la visualización. Objeto de conferencia, Universidad Nacional de La Plata, Buenos Aires. Obtenido de <http://sedici.unlp.edu.ar/handle/10915/21774>
- García Tenorio, Á. (2019). Detección de Puntos Significativos en Imágenes para Reconstrucción 3D . Trabajo de Fin de Grado, Universidad Complutense de Madrid, Madrid. Obtenido de <https://eprints.ucm.es/56694/>
- Grayson, J. E. (2000). Python and Tkinter Programming. Manning.
- Isaza, J. F., & Correa, S. (2008). Metodología para la reconstrucción 3D de estructuras craneofaciales y su utilización en el método de elementos finitos. Artículo, Universidad EAFIT, Medellín. Obtenido de [https://www.researchgate.net/publication/28232753\\_Metodologia\\_para\\_la\\_reconstruccion\\_3D\\_de\\_estructuras\\_craneofaciales\\_y\\_su\\_utilizacion\\_en\\_el\\_metodo\\_de\\_elementos\\_finitos](https://www.researchgate.net/publication/28232753_Metodologia_para_la_reconstruccion_3D_de_estructuras_craneofaciales_y_su_utilizacion_en_el_metodo_de_elementos_finitos)
- Martínez Walker, C. P. (2016). Digitalización a bajo costo de objetos religiosos usando reconstrucción tridimensional con la luz blanca. Trabajo de grado, Universidad Nacional de Colombia, Bogotá. Obtenido de <https://repositorio.unal.edu.co/handle/unal/59916>
- Mercado Anguilla, L. M. (2018). Procesamiento y Reconstrucción tridimensional de estructuras cardiacas a partir de imágenes de TAC. Trabajo de grado, Univeridad del Norte, Barranquilla. Obtenido de <http://manglar.uninorte.edu.co/handle/10584/8023>
- Morato Toro, R. (2005). Estudio de un sistema de reconstrucción 3D en tiempo real. Proyecto final de carrera, Universidad de Sevilla, Sevilla. Obtenido de <http://bibing.us.es/proyectos/abreproy/11057/>
- Moreno Fernández, S. (2020). Herramienta de Reconocimiento de Imágenes en Python. Proyecto fin de carrera, Universidad de Sevilla, Sevilla. Obtenido de <https://idus.us.es/handle/11441/102090>
- Muñoz, L., Quiñónez, E., & Victoria, H. (2013). Reconstrucción 3D de objetos sumergidos en aguas limpias. Artículo de Investigación, Universidad Distrital Francisco José de Caldas. Obtenido de <http://hdl.handle.net/11349/19786>
- Palos Cuesta, J. (2012). Desarrollo de un procedimiento para la creación de imágenes 3D a partir de imágenes 2D. Trabajo final de carrera, Universidad Politecnica de Valencia, Gandia. Obtenido de <https://riunet.upv.es/handle/10251/17762>
- Ramos Howell, P. A. (2019). Desarrollo de un sistema de reconstrucción 3D a partir de imágenes capturadas por drones. Trabajo fin de grado, Universidad Autonoma de Madrid, Madrid. Obtenido de <http://hdl.handle.net/10486/689090>
- Relevo Luna, D. A., Usama, F. D., & Flórez Marulanda, J. F. (2012). Reconstrucción 3D de escenas mediante un sistema de visión estéreo basado en extracción de características y desarrollado en OpenCV. Artículo, Universidad del Cauca, Popayán. Obtenido de [http://www.scielo.org.co/scielo.php?script=sci\\_abstract&pid=S0123-21262012000200011&lng=es](http://www.scielo.org.co/scielo.php?script=sci_abstract&pid=S0123-21262012000200011&lng=es)
- Soriano Marín, C. (2012). Síntesis de vistas a partir de imágenes e información 3D. Proyecto fin de carrera, Universidad de Zaragoza, Zaragoza. Obtenido de <http://zaguan.unizar.es/record/4586?ln=es>
- Tibaquirá Artunduaga, E. J. (2020). Reconstrucción 3D de rostros a partir de una secuencia multimodal para el diseño de ascaras de apnea de sueño. Trabajo de fin de grado, Barcelona. Obtenido de <http://hdl.handle.net/2117/181982>
- Wong Quijano, J. O. (s.f.). Reconstrucción de objetos mediante cámaras estéreo, utilizando múltiples vistas. Tesis, Universidad Veracruzana, Veracruz. Obtenido de <https://cdigital.uv.mx/bitstream/handle/123456789/46841/WongQuijanoJesus.pdf;sequence=2>

(1) Organización para la Cooperación y Desarrollo Económico (OCDE)

(2) PA: Plan de Aula, PI: Proyecto integrador, TG: Trabajo de Grado, RE: Reda

#### **AUTORIZACIÓN**

Al diligenciar este documento, autorizo de manera previa, expresa e inequívoca a UNIDADES TECNOLÓGICAS DE SANTANDER a dar tratamiento de mis datos personales aquí consignados, conforme a la autorización otorgada (por mi o por mi representante legal) al momento de celebrada la matrícula, incluyendo el consentimiento explícito para tratar datos sensibles aun conociendo la posibilidad de oponerme a ello, conforme a las finalidades incorporadas en la Política de Tratamiento de Información publicada en [www.uts.edu.co](http://www.uts.edu.co) y/o en Calle de los estudiantes 9-82 Ciudadela Real de Minas, que declaro conocer y estar informado que en ella se presentan los derechos que me asisten como titular y los canales de atención donde ejercerlos.