

DEVELOPMENT OF A SYSTEM FOR TRACKING OBJECTS IN SPACE AND DETERMINING THEIR TRAJECTORS

DESARROLLO DE UN SISTEMA DE SEGUIMIENTO DE OBJETOS EN EL ESPACIO Y DETERMINACIÓN DE SUS TRAYECTORIAS

Ing. Andrés David Cohen Gamarra, PhD. José Orlando Maldonado B

Universidad de Pamplona

Grupo de Investigación en Ciencias Computacionales – CICOM, Programa de Ingeniería de Sistemas.

Tel.: 57-7-5685303, Fax: 57-7-5685303, Ext. 144

E-mail: {andres.cohen, orlmaldonado}@unipamplona.edu.co

Abstract: This document addresses the development and implementation of a system for monitoring objects, location and trajectory generation in a three-dimensional space, through computer vision and artificial intelligence techniques, by which they will be studied and selected the recent tools, techniques and algorithms found in the bibliography available, in order to select and adapt the most suitable for the achievement of draft. Based on the basic methods, algorithms are developed to solvetion to the established requirements. The resulting algorithms will be embedded in a functional prototype that facilitates its operation through a graphical interface that allows Check the configuration of the tracking camera system.

The system will be evaluated in different controlled scenarios to determine the error of tracking, location and trajectory generation. The set of algorithms developed Rolled up will be the basis for future applications in specific problems.

Keywords: Object detection, Object tracking, Stereo Vision, video analysis.

Resumen: Este documento aborda el desarrollo e implementación de un sistema para el seguimiento de objetos, ubicación y generación de trayectorias en un espacio tridimensional, mediante técnicas de visión por computador e inteligencia artificial, para ello se estudiarán y seleccionarán las herramientas, técnicas y algoritmos recientes encontrados en la bibliografía disponible, con el objeto de seleccionar y adaptar los más adecuados para la consecución del proyecto. A partir de los métodos de base, se desarrollan los algoritmos que den solución a los requerimientos establecidos. Los algoritmos resultantes serán embebidos en un prototipo funcional que facilite su operación mediante una interfaz gráfica que permite la configuración del sistema de cámaras de seguimiento.

El sistema será evaluado en diferentes escenarios controlados para determinar el error de seguimiento, ubicación y generación de trayectorias. El conjunto de algoritmos desarrollados serán base para futuras aplicaciones en problemas específicos.

Palabras clave: Detección de objetos, Seguimiento de objetos, Visión Estéreo, análisis de video.

1. INTRODUCCIÓN

Para poder conocer realmente el mundo donde vivimos lo hacemos mirando, así es que llegamos a conocer el mundo en el que vivimos. El

entorno que nos rodea está lleno de infinitos tipos de objetos impresionantes.(Xiang et al. 2016). En otras palabras la visión es un medio por el cual se puede comprender el ambiente que

nos rodea.(Yoon et al., n.d.). Al introducir el termino visión en la computación se puede decir que es un proceso donde las computadoras intentan emular la capacidad de un ser vivo al analizar imágenes, videos entendiendo estas y actuando en consecuencia con un alto nivel de comprensión.(Enrique Sucar and Giovanni Gómez, n.d.). El continuo incremento de nuevos algoritmos y aplicaciones hacen de esta una disciplina en constante evolución que a través de las ultimas décadas ha experimentado un rápido avance, éste crecimiento se puede evidenciar en las numerosas investigaciones y publicaciones existentes en la comunidad científica.(Muñoz, López, and Caicedo 2008).

Este incremento puede deberse a el gran contenido visual (imágenes y videos) que se ha generado en la actualidad y que se estará incrementando a medida pasan los años, así también se puede deber a el almacenamiento y procesamiento de los dispositivos que tenemos a mano , librerías y lenguajes de programación para dicho fin.(Marcos, Ana. Martines, francisco. pernia , Alpha. Elias, Fernando. Castejón, Manuel. Ordieres, Joaquín. Vergara, n.d.)El seguimiento de objetos es un área dentro de la visión por computadora o visión artificial que tiene muchas aplicaciones prácticas tales como (video vigilancia, interacción humano-computadora, Análisis de flujo vehicular, análisis del movimiento corporal, comportamiento de grupos de personas, sistemas de video vigilancia, etc.).(Matteo and Gómez 2014).El problema que se presenta al hacer el seguimiento de objetos en video puede resumirse como la tarea de encontrar la posición de un objeto en cada cuadro o frame. La capacidad que se requiere para poder rastrear un objeto en frames consecutivos (video) depende de muchos factores, algunos de estos son: el conocimiento que se tiene sobre el objeto, el tipo de parámetros a rastrear y el video que muestra el objeto.(RAFAEL ORMAECHEA IZQUIERDO and .SANTIAGO AJA FERNÁNDEZ 2015) .Al ser esta una disciplina tan compleja involucra a otras ciencias como lo son la física, matemáticas entre otras. la visión artificial es una herramienta que tiene la capacidad de establecer una relación entre el mundo tridimensional y las vistas bidimensionales que se toman.(Cabello 2009). Por la cual se puede hacer uso de la visión estéreo ya que a partir de esta herramienta se pueden formar imágenes tridimensionales estableciendo una relación entre el mundo tridimensional y las vistas bidimensionales de dos cámaras Permitiendo reconocer la profundidad de una imagen y realizar mediciones en la imagen.(Wainschenker Mg Ing José María Massa Mg Ing Paula Tristan, n.d.).

Haciendo localizaciones en 3D. Entre otras cosas, los puntos de coincidencia entre las dos

cámaras se puede encontrar haciendo la calibración de las cámaras y encontrando la matriz intrínseca resultante y después aplicar geometría epipolar para obtener la distancia a la que se encuentra el objeto .

En la visión estéreo, se realizan estos cinco pasos:

1. Eliminación de la distorsión radial y tangencial a través de cálculos matemáticos para obtener imágenes sin distorsiones.

2. Rectificación del ángulo y el espaciado de las imágenes. Esta etapa permite imágenes coplanares en el eje Y, esto ayudará a encontrar correspondencia. En este proyecto se utilizo el algoritmo de Bouguet.

3. Encuentre la misma característica en la imagen derecha e izquierda. Entonces obtienes un mapa de disparidad que muestra las diferencias entre las imágenes en el eje X.

4. La triangulación. El mapa de disparidad se transforma en distancias por triangulación.

5. Reconstrucción de la escena 3D esta establece una correspondencia 3D a partir de las proyecciones 2D.

2. MÉTRICAS ESTÁNDAR MÁS UTILIZADAS EN MODELOS DE REGRESIÓN.

Root Mean Squared Error

Sintaxis RMSE(X, F):

X es el resultado eventual de datos de las series de tiempo.

F es el pronóstico de datos de series de tiempo.

El error cuadrático medio (RMSE) es la desviación estándar de los residuos (errores de predicción). Los residuos son una medida de qué tan lejos están los puntos de datos de la línea de regresión; RMSE es una medida de la dispersión de estos residuos. En otras palabras, le dice qué tan concentrados están los datos alrededor de la línea de mejor ajuste. El error cuadrático medio se usa comúnmente en climatología, predicción y análisis de regresión para verificar resultados experimentales(Anthony G, n.d.).

La fórmula seria la siguiente:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (A_t - F_t)^2}{n}}$$

Dónde:

A_t son las observaciones

F_t pronostica los valores de una variable.

n el número de observaciones disponibles para el análisis.

Mean Absolute Error

Sintaxis MAE(X, Y)

X son los datos originales de la muestra de series de tiempo.

Y son los datos de las series de tiempo pronosticadas.

El error absoluto medio es una medida de evaluación del modelo utilizado con los modelos de regresión.(Sammut and Webb 2010)El error absoluto medio de un modelo con respecto a un conjunto de prueba es la media de los valores absolutos de los errores de predicción individuales en todas las instancias del conjunto de prueba. Cada error de predicción es la diferencia entre el valor verdadero y el valor predicho para la instancia (Ait-Amir, Pougnet, and El Hami 2015)

La fórmula sería la siguiente:

$$MAE = \frac{\sum_{t=1}^n |A_t - F_t|}{n}$$

Dónde:

A_t Son las observaciones actuales de las series de tiempo

F_t Es la serie de tiempo estimada o pronosticada

n Es el número de puntos de datos no faltantes

VENTAJAS Y DESVENTAJAS DE LOS ALGORITMOS DE SEGUIMIENTO ESTUDIADOS

- **2.1 Tracking based color:** El color ha sido ampliamente utilizado en sistemas de seguimiento en tiempo real. Ya que estos ofrecen varias ventajas significativas sobre señales geométricas como la simplicidad computacional, la robustez bajo oclusión parcial, rotación, escala y cambios de resolución. Aunque los métodos de color resultaron ser eficiente en una variedad de aplicaciones de visión, hay varios

problemas asociados con estos métodos en los que la constancia del color es uno de los más importantes.

Ventajas

- El sistema rastrea objetos de colores con buena precisión.
- Permite rastrear objetos en video en vivo.

Desventajas

- Baja precisión en imágenes de cámara de baja resolución.
- Baja precisión en condiciones de poca luz.

- **2.2 BOOSTING Tracker:** Este rastreador se basa en una versión en línea de AdaBoost, el algoritmo que el detector facial basado en cascada HAAR usa internamente. Este clasificador debe ser entrenado en Tiempo de ejecución con ejemplos positivos y negativos del objeto. El cuadro delimitador inicial suministrado por el usuario (o por otro algoritmo de detección de objetos) se toma como un ejemplo positivo para el objeto, y muchos parches de imagen fuera del cuadro delimitador se tratan como fondo. Dado un nuevo marco, el clasificador se ejecuta en cada píxel en la vecindad de la ubicación anterior y se registra la puntuación del clasificador. La nueva ubicación del objeto es aquella donde la puntuación es máxima. Entonces ahora tenemos un ejemplo positivo más para el clasificador. A medida que entran más cuadros, el clasificador se actualiza con estos datos adicionales(“Seguimiento de Objetos Usando OpenCV (C ++ / Python) | Aprende OpenCV” n.d.).

Ventajas

- El rastreador se utiliza para hacer comparaciones.
- El rastreador utiliza herencia.

Desventajas

- El rastreador es muy lento.
- El rastreador no funciona muy bien.

- **2.2 MIL Tracker:** Este rastreador es similar en idea al rastreador BOOSTING descrito anteriormente. La Gran diferencia es que, en lugar de considerar solo la ubicación actual del objeto como un ejemplo positivo, se ve en un pequeño vecindario alrededor de la ubicación actual para generar varios ejemplos positivos potenciales. Puede

estar pensando que es una mala idea porque en la mayoría de estos ejemplos”positivos. El objeto no está centrado. Aquí es donde el aprendizaje de instancias múltiples (MIL) viene a rescatar. En MIL, no se especifican ejemplos positivos y negativos, sino”bolsas” positivas y negativas (“Seguimiento de Objetos Usando OpenCV (C ++ / Python) | Aprende OpenCV” n.d.). La colección de imágenes en la bolsa positiva no son todos ejemplos positivos. En cambio, ¡solo una imagen en la bolsa positiva debe ser un ejemplo positivo! (“Milk” n.d.)

Ventaja

- El rendimiento es bastante bueno. No se desplaza tanto como el rastreador BOOSTING y hace un trabajo razonable bajo oclusión parcial.

Desventaja

- La falla de seguimiento no se informa de manera confiable. No se recupera de la oclusión completa.

• **2.3 KCF Tracker**

KFC significa filtros de correlación kernelized . Este rastreador se basa en las ideas presentadas en los dos rastreadores anteriores. Este rastreador utiliza el hecho de que las múltiples muestras positivas utilizadas en el rastreador MIL tienen grandes regiones superpuestas. Estos datos superpuestos conducen a algunas buenas propiedades matemáticas que este rastreador aprovecha para hacer que el seguimiento sea más rápido y preciso al mismo tiempo.(Mallick n.d.)

Ventaja

- La precisión y la velocidad son mejores que MIL e informa de un error de seguimiento mejor que BOOSTING y MIL.

Desventaja

- No maneja bien la oclusión completa.

• **2.4 CSRT Tracker**

Es un Filtro de correlación discriminativa con canal y confiabilidad espacial (DCF-CSR), utilizamos el mapa de confiabilidad espacial para ajustar el soporte del filtro a la parte de la región seleccionada desde el marco para el seguimiento. Esto asegura la

ampliación y localización de la región seleccionada y un seguimiento mejorado de las regiones u objetos no rectangulares. Utiliza solo 2 características estándar (HoGs y Colornames). También funciona a fps comparativamente más bajos (25 fps) pero ofrece una mayor precisión para el seguimiento de objetos.(Mallick n.d.)

Ventaja

- Es más preciso que KCF.

Desventajas

- Más lento que KFC.

• **2.5 Motion detection**

Ventajas

- Es un método bastante robusto a los cambios de iluminación.
- Es un método bastante robusto a los cambios de movimiento.
- Consigue estabilizarse pasado un tiempo.

Desventajas

- Si el objeto o la persona en movimiento se quedan quietos, no se detecta.
- No es capaz de detectar siluetas

• **2.6 SSD single shot detection**

Es un algoritmo popular en la detección de objetos. Generalmente es más rápido que Faster RCNN

Ventajas

- Es un método bastante ágil.
- Es un método eficiente basado en el aprendizaje profundo.
- Modelo fácil de entrenar.
- Modelo de fácil integración.

Desventaja

- Pocas etiquetas en la clase MobileNets.

3EXPERIMENTACIÓN DE LOS ALGORITMOS SELECCIONADOS

3.1 Algoritmos de seguimiento

Seguimiento múltiple de objetos normal: Después de las 7 pruebas realizadas del algoritmo de Color y CSRT los resultados obtenidos fueron óptimos para el caso del algoritmo CSRT el cual sigue el objeto el 100 % de las veces y el algoritmo de detección de color sigue el objeto un 89.53 % de las veces. En la **Figura 1** se puede observar los resultados obtenidos en el algoritmo CSRT y en la **Figura 2**

se puede observar los resultados obtenidos en el algoritmo de color. El porcentaje de error fue calculado a través del error relativo porcentual.

Lugar: Habitación.	Descripción: El video se graba en una habitación controlada con un fondo blanco
Iluminación: iluminación artificial, situados en el techo de la habitación,	Altura de la cámara: La cámara se posiciona a 50 cm sobre el suelo.

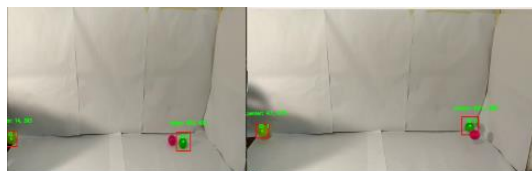


Figura 2: Prueba del algoritmo CSRT con pelotas de colores



Figura 1: Prueba del algoritmo de color

Número de pruebas	Precisión de pronóstico de seguimiento	Numero de objetos	Color objetos	Diámetro de objetos en (CM)	Error
Prueba 1	100%	3	Amarillo, Azul, Rojo	9	0%
Prueba 2	100%	4	Verde, Blanco, Azul, Rojo	9	0%
Prueba 3	100%	3	Verde, Azul, Blanco	Variado 1 obj. 5 2 obj. 9	0%
Prueba 4	100%	2	Azul, Rojo	9	0%
Prueba 5	100%	5	Verde, Blanco, Amarillo, Rojo, Azul	Variado 2 obj. 5 3 obj. 9	0%
Prueba 6	100%	2	Azul, Amarillo	9	0%
Prueba 7	100%	3	Rojo, Azul, Amarillo	Variado	0%

Tabla 1 Porcentaje de error del algoritmo CSRT Seguimiento múltiple de objetos.(Normal)

Número de pruebas	Precisión de pronóstico de seguimiento	Numero de objetos	Color objetos	Diámetro de objetos en (CM)	Error
Prueba 1	100%	3	Amarillo, Azul, Rojo	9	0%
Prueba 2	100%	4	Verde, Blanco, Azul, Rojo	9	0%
Prueba 3	66.6%	3	Verde, Azul, Blanco	Variado 1 obj. 5 2 obj. 9	33.3%
Prueba 4	100%	2	Azul, Rojo	9	0%
Prueba 5	60%	5	Verde, Blanco, Amarillo, Rojo, Azul	Variado 2 obj. 5 3 obj. 9	40%
Prueba 6	100%	2	Azul, Amarillo	9	0%
Prueba 7	100%	3	Rojo, Azul, Amarillo	9	0%

Tabla 2 Porcentaje de error del algoritmo de color Seguimiento múltiple de objetos. (Normal)

Seguimiento de un objeto (prueba ping pong):

Con esta respectiva prueba se pretende probar el código ante una experiencia más exigente, donde se batea la pelota contra la pared en diferentes ocasiones como se muestra en la **Figura 3**. Al igual que en la situación anterior los resultados obtenidos fueron excelente cumpliéndose con un seguimiento del 100 % lo que demuestra la solidez del seguimiento, La prueba del algoritmo de color tuvo un error del 100% ya que no siguió el objeto y se puede ver el resultado en la **Figura 4**. El porcentaje de error fue calculado a través del error relativo porcentual.

Lugar: Habitación.	Descripción: El video se graba en una habitación controlada con un fondo blanco
Iluminación: iluminación artificial, situados en el techo de la habitación,	Altura de la cámara: la cámara se posiciona a 50 cm sobre el suelo.

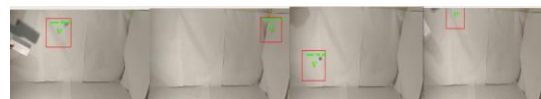


Figura 3: Prueba del algoritmo CSRT con pelota de ping pong

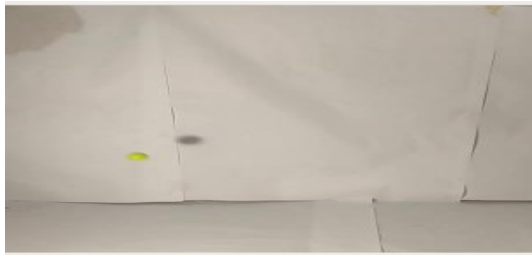


Figura 4: Prueba del algoritmo de color con pelota de ping pong

Número de pruebas	Precisión de pronóstico de seguimiento	Numero de objetos	Color objetos	Diámetro de objetos en (CM)	Error
Prueba 1	100%	1	Blanco	5	0%
Prueba 2	100%	1	Blanco	5	0%
Prueba 3	100%	1	Blanco	5	0%
Prueba 4	100%	1	Blanco	5	0%
Prueba 5	100%	1	Blanco	5	0%
Prueba 6	100%	1	Blanco	5	0%
Prueba 7	100%	1	Blanco	5	0%

Tabla 3 Porcentaje de error del algoritmo CSRT Seguimiento de un objeto (Prueba ping pong)

Número de pruebas	Precisión de pronóstico de seguimiento	Numero de objetos	Color objetos	Diámetro de objetos en (CM)	Error
Prueba 1	0%	1	Blanco	5	100%
Prueba 2	0%	1	Blanco	5	100%
Prueba 3	0%	1	Blanco	5	100%
Prueba 4	0%	1	Blanco	5	100%
Prueba 5	0%	1	Blanco	5	100%
Prueba 6	0%	1	Blanco	5	100%
Prueba 7	0%	1	Blanco	5	100%

Tabla 4 Porcentaje de error del algoritmo de color Seguimiento de un objeto (Prueba ping pong)

3. DETERMINAR LA DISTANCIA DE LAS CÁMARAS AL OBJETO (TRIANGULACIÓN)

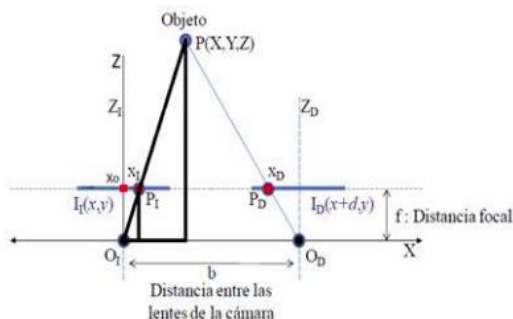


Figura 5: Geometría epipolar

Después de que se haya generado el mapa de disparidad, se debe determinar la distancia. El trabajo es encontrar la relación entre el valor de disparidad y la distancia. Para hacer esto hemos medido los valores de disparidad

experimentalmente en varios puntos para determinar una regresión como se ve en la Figura 6

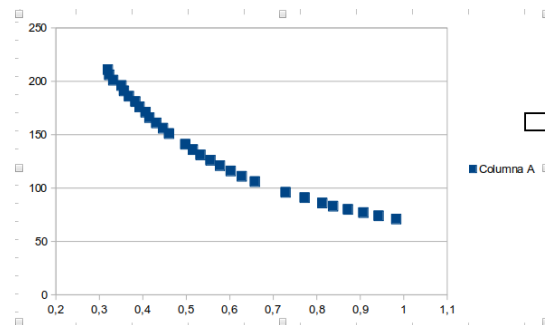


Figura 6: Distancia contra disparidad

Ecuación de línea recta en el programa Python average es la representación de X

$$\text{Distance} = -469.67245074286 * (\text{average}) ** 3 + 1207.65108848869 * (\text{average}) ** 2 - 1135.03367515956 * (\text{average}) + 462.731877414055$$

Figura 7: Ecuación de línea recta

Para obtener esta ecuación de regresión en línea recta, se utilizó la capa de empaque openpyxl para guardar los valores de la disparidad en un archivo de Excel.

La medición de distancia solo es válida desde una distancia de 67 cm a 203 cm para obtener buenos resultados. La precisión de la medición también depende de la calidad de la calibración. Con nuestras cámaras estéreo están a una distancia de un objeto con una precisión de +/- 3 cm

4. GENERACION DE LAS COORDENADAS 3D

Se establecen las coordenadas 3D a partir de las proyecciones 2D de la matriz rectificadas, donde las imágenes son coplanares, a través de la **Figura 5** se obtuvieron las coordenadas X,Y a través de semejanza de triángulos.

Donde:

$$(1) \quad \frac{X}{Z} = \frac{x_I - x_O}{f}$$

Despejando a X se obtiene:

$$(2) \quad X = \frac{Z}{f} (x_I - x_O)$$

$$(3) \quad Y = \frac{z}{f} (y_I - y_o)$$

La ecuación (2) representa la coordenada X del punto P en las coordenadas 3D y La ecuación (3) representa la coordenada Y del punto P en las coordenadas 3D.

Donde Z hace referencia a la distancia obtenida anteriormente, la f es la distancia focal de las cámaras la cual es igual para ambas y esta viene dada en unidades de pixel, y_I es la coordenada Y del centroide en la imagen izquierda e y_o es la coordenada Y del punto principal de la imagen, X_o es la coordenada X del punto principal de la imagen, X_I es la coordenada del objeto en el eje X de la cámara izquierda.

Se debe tener cuidado con las unidades de cada parámetro esto con el fin de que las unidades correspondan a las distancia del objeto. Teniendo las fórmulas (2),(3) ya se puede calcular las coordenadas de todos los puntos de la escena tridimensional, es decir, podemos conocer la posición del objeto en cada instante de tiempo, teniendo entonces el seguimiento del objeto y las coordenadas del mismo completamente definidas se puede representar cada punto en un gráfico.

5. EXPERIMENTACIÓN

El procedimiento realizado en este trabajo se puede describir en la Figura. Las imágenes estéreo se producen utilizando dos cámaras a una cierta distancia entre ellas y varia entre 10cm a 30cm. en el trabajo realizado se utiliza una distancia de 11 cm entre las cámaras. Con un focal de 3.85 mm especificado en las cámaras.

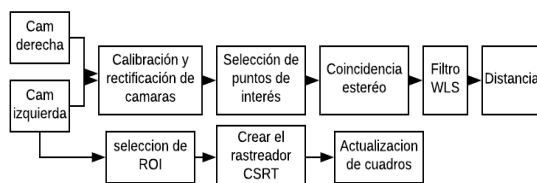


Figura 8: Procedimiento del algoritmo

La calibración de la cámara se usa para garantizar que ambas cámaras están en una posición paralela perfecta y para eliminar la distorsión. Esta calibración se basa mediante el uso de Imgpoints y se usa la calibración estándar la cual se realiza a través de un tablero de ajedrez, guardando los puntos de la posición de las esquinas del tablero de ajedrez. La Figura 9 muestra el proceso de calibración y en la Figura 10 se muestran las líneas epipolares



Figura 10: Proceso de calibración de cámaras

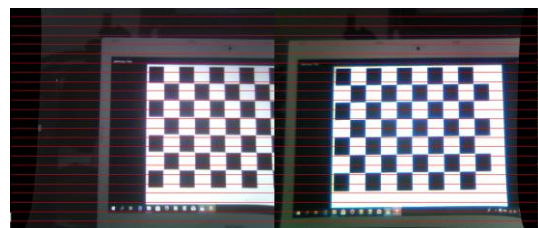


Figura 9: Líneas epipolares con la cámara calibrada

Después de haber realizado la calibración, ambas cámaras están listas para usarse. La matriz que fue creada por el proceso de calibración de las cámaras es necesaria para eliminar la distorsión radial y tangencial de las cámaras cámara y encontrar las líneas epipolares en ambas imágenes. El paso que sigue es calcular el mapa de disparidad, para ello se utiliza un algoritmo de coincidencia de bloque semi-global (SGBM) para una correspondencia estéreo entre las imágenes de la cámara derecha y la cámara izquierda.

Para encontrar la disparidad, se restan las coordenadas del bloque de coincidencia con el bloque de referencia, entonces el valor absoluto se toma del resultado y cuanto mayor sea este valor más cerca esta el objeto de la cámara estéreo.

Para el algoritmo de coincidencia estéreo se coloco el objeto a una distancia de 70-211 cm de las cámaras y capturando la disparidad en cada punto como se muestra en la Figura 11.

Para ello se coloco una cinta métrica para medir la distancia entre las cámaras y el objeto, cuyo proceso se realiza haciendo doble clic en el objeto en la pantalla de la computadora. Estos datos se escriben en un archivo Excel.



Figura 11: Medición de distancia experimental

Después de aplicar el algoritmo SGBM se aplica el filtro WLS que se utiliza para reducir el ruido en las imágenes. La **Figura 12** muestra el resultado del filtro WLS.

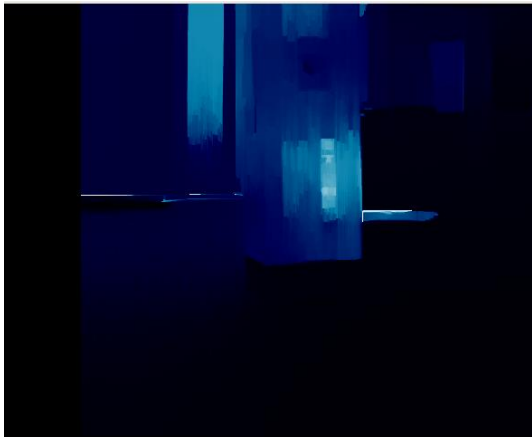


Figura 12: Filtro WSL aplicado al resultado de SGBM

Calculadas las coordenadas X, Y con las formulas (2) y (3) mostradas anteriormente y para cada instante de tiempo y teniendo la coordenada Z calculada anteriormente también para cada instante, se realizar la reconstrucción de la escena 3D del objeto. En este caso, habrá que considerar los ejes que utiliza Python para la representación grafica de los puntos. y como el parámetro Z representa la distancia que hay de la cámara al objeto, el eje Z no corresponde, por lo tanto se intercambian los ejes de la siguiente forma:
Se intercambia el eje Y por el eje Z y el eje Z por el eje Y.

Aplicando el seguimiento CSRT y la visión estéreo obtuvimos las siguientes trayectorias 3D, como se muestra en la **Tabla 5**, **Tabla 6**, **Tabla 7**.

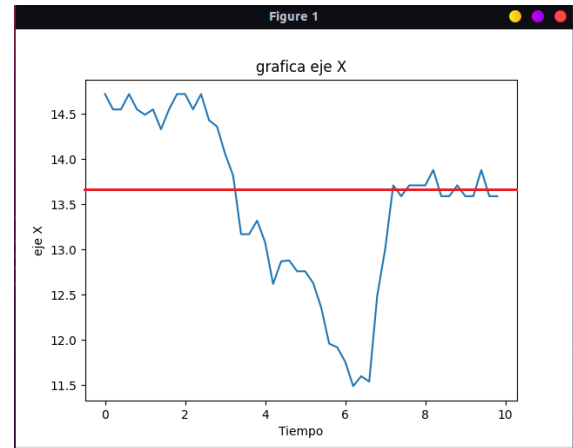


Tabla 5 Grafica de los puntos 3D del eje X obtenidos

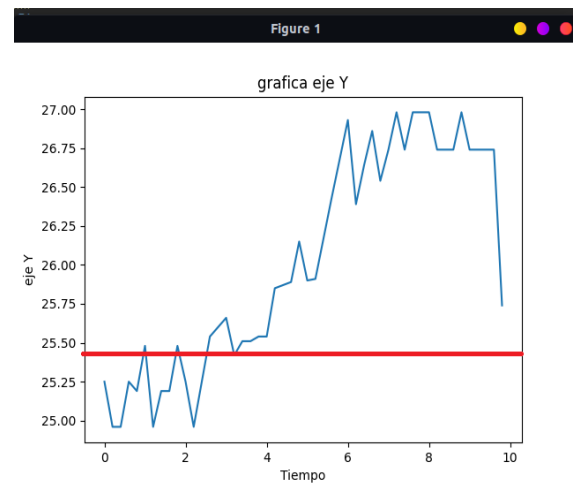


Tabla 6 Grafica de los puntos 3D del eje Y obtenidos

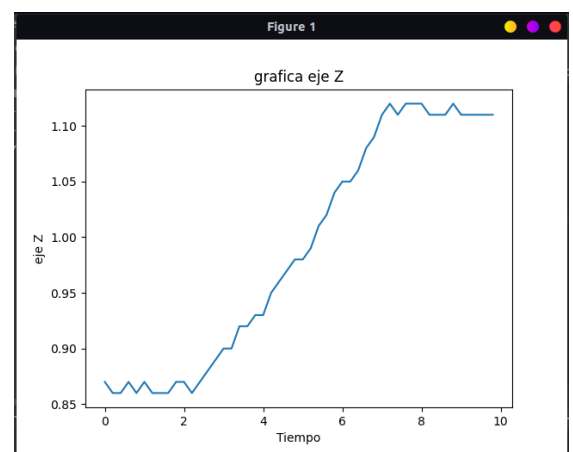


Tabla 7 Grafica de los puntos 3D del eje Z obtenidos

El seguimiento se hizo en un tiempo de 10 segundos, al iniciar el seguimiento hubo un retraso del segundo cero al segundo 2, y como el objeto llego antes al punto establecido en el

segundo 7, este espero y mantuvo una constancia del segundo 7 al segundo 10. En la gráfica X y en la gráfica Y se presentó un error debido a la calibración de las cámaras.

5. PORCENTAJE DE ERROR DEL SEGUIMIENTO A TRAVES DE LAS METRICAS MAE, MSE, RMSE.

Tiempo	Actual	Modelo	Error	Valor del error Absoluto	Error cuadrado
t	A_t	F_t	$A_t - F_t$	$ A_t - F_t $	$(A_t - F_t)^2$
0	13,600	14,800	-1,200	1,200	1,440
1	13,600	14,400	-0,800	0,800	0,640
2	13,600	13,300	0,300	0,300	0,090
3	13,600	14,000	-0,400	0,400	0,160
4	13,600	13,400	0,200	0,200	0,040
5	14,000	12,280	1,720	1,720	2,958
6	14,000	11,500	2,500	2,500	6,250
7	14,000	13,600	0,400	0,400	0,160
8	14,800	13,700	1,100	1,100	1,210
9	13,600	13,700	-0,100	0,100	0,010
10	13,600	13,500	0,100	0,100	0,010
Total			3,820	8,820	12,968
n			11		
Métrica			% de error		
MAE			0,802		
MSE			1,179		
RMSE			1,086		

Tabla 8 Porcentaje de error calculado por las métricas estudiadas en el eje X

Tiempo	Actual	Modelo	Error	Valor del error Absoluto	Error cuadrado
t	A_t	F_t	$A_t - F_t$	$ A_t - F_t $	$(A_t - F_t)^2$
0	25,25	25,25	0,000	0,000	0,000
1	25,32	25,5	-0,180	0,180	0,032
2	25,35	25	0,350	0,350	0,123
3	25,49	25,49	0,000	0,000	0,000
4	25,52	25,52	0,000	0,000	0,000
5	25,28	26,23	-0,950	0,950	0,902
6	26,32	26,89	-0,570	0,570	0,325
7	26,5	26,5	0,000	0,000	0,000
8	26,8	26,89	-0,090	0,090	0,008
9	26,89	26,89	0,000	0,000	0,000
10	26,89	25,75	1,140	1,140	1,300
Total			-0,300	3,280	2,690
n			11		
Métrica			% de error		
MAE			0,298		
MSE			0,245		
RMSE			0,495		

Tabla 9 Porcentaje de error calculado por las métricas estudiadas en el eje Y

Tiempo	Actual	Modelo	Error	Valor del error Absoluto	Error cuadrado
t	A_t	F_t	$A_t - F_t$	$ A_t - F_t $	$(A_t - F_t)^2$
0	0,860	0,860	0,000	0,000	0,000
1	0,860	0,870	-0,010	0,010	0,000
2	0,680	0,860	-0,180	0,180	0,032
3	0,900	0,900	0,000	0,000	0,000
4	0,950	0,950	0,000	0,000	0,000
5	1,000	1,000	0,000	0,000	0,000
6	1,050	1,050	0,000	0,000	0,000
7	1,100	1,120	-0,020	0,020	0,000
8	1,100	1,100	0,000	0,000	0,000
9	1,100	1,100	0,000	0,000	0,000
10	1,100	1,100	0,000	0,000	0,000
Total			-0,210	0,210	0,033
n			11		
Métrica			% de error		
MAE			0,019		
MSE			0,003		
RMSE			0,055		

Tabla 10 Porcentaje de error calculado por las métricas estudiadas en el eje Z

Las dimensiones están dadas en centímetros, la variación que se presenta en la Tabla 5 y Tabla 6 se debe a que el objeto se va alejando y el objeto de interés se va haciendo más pequeño y debido a eso el centroide queda colocado a una altura mayor. Por otra parte se realizó una comparación de las métricas estudiadas para verificar cuanto porcentaje de error calculaba cada modelo y si un modelo tenía una diferencia mayor a los otros. Obteniendo estos resultados:

Porcentaje de error para el eje X:

- MAD 0,802
- MSE 1,179
- RMSE 1,086

Porcentaje de error para el eje Y

- MAD 0,298
- MSE 0,245
- RMSE 0,495

Porcentaje de error para el eje Z

- MAD 0,019
- MSE 0,003
- RMSE 0,055

Donde el mayor porcentaje de error fue del modelo MSE en el eje X y podemos observar que el eje que produce menos porcentaje de error es el eje Z el cual es el eje en el cual se está observando la distancia medida del objeto seguido.

6. ERROR DEL MODELO FINAL CON LA MÉTRICA MAE.

Métrica MAE	Eje X	Eje Y	Eje Z
Prueba 1	0,349	0,748	0,021
Prueba 2	0,325	0,823	0,012
Prueba 3	0,298	0,802	0,019
Prueba 4	0,273	0,866	0,034
Total error	0,31125	0,80975	0,0215

El porcentaje de error calculado para las pruebas finales tienen un valor mínimo, por lo tanto se puede concluir que los valores calculados por el modelo se aproximan a los valores del mundo real.

Se puede usar el modelo con el cual se sienta más cómodo, ya que los tres hacen lo mismo. El criterio MAE es un criterio de medida similar al RMSE y al MSE. Sin embargo, es más robusto ya que es menos sensible a los valores extremos que MSE Y RMSE. Todas las mediciones de distancia (MSE, RMSE y MAE) son equivalentes y ayudan a cuantificar la precisión de la solución aproximada en comparación con los datos simulados. Un valor pequeño para estos criterios significa que el modelo estimado está cerca del valor real.

7. CONCLUSIONES

Luego de analizar los métodos seleccionados se realizaron las pruebas en la cuales se tuvieron en cuenta las complicaciones que se presentaron durante el desarrollo del trabajo de grado, como la claridad, ruido, sobre-posición, seguimiento y rapidez. Dando como resultado un algoritmo eficiente y preciso donde el promedio ponderado de error al medir la distancia es de 1.1425 y este puede aumentar a 2 % de error, en cuanto al seguimiento, este es capaz de seguir el objeto a un 100 % mientras no presente oclusión parcial total, el flujograma del algoritmo se presenta en la **Figura 8**.

El seguimiento realizado al objeto ayuda en gran medida a la reconstrucción de la escena 3D sin embargo se producen errores de trayectoria del objeto seguido. Esto puede ser por los siguientes factores:

1. La calibración estereoscópica, ya que esta ayuda a disminuir el error.

2. El uso del filtro WLS, que ayuda a tener una disparidad del objeto mas aproximada a la real ya que no abra tanto ruido en la imagen.

REFERENCIAS

- Ait-Amir, Bouzid, Philippe Pougnet, and Abdelkhalak El Hami. 2015. "Meta-Model Development." *Embedded Mechatronic Systems 2*: 151–79. <https://doi.org/10.1016/b978-1-78548-014-0.50006-2>.
- Anthony G, Barnston. n.d. "Correspondence among the Correlation, RMSE, and Heidke Forecast Verification; Refinement of the Heidke Score."
- Cabello, David Jiménez. 2009. "Sistema de Monitorización Mediante Visión Artificial de Zonas de Riesgo Potencial Para La Circulación Ferroviaria y Seguridad de Las Personas."
- Enrique Sucar, L, and México Giovanni Gómez. n.d. "Visión Computacional."
- Mallick, Satya. n.d. "Object Tracking Using OpenCV (C++/Python) | Learn OpenCV." Accessed March 1, 2020. <https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/>.
- Marcos, Ana. Martinez, Francisco. Pernía, Alpha. Elias, Fernando. Castejón, Manuel. Ordieres, Joaquín. Vergara, Eliseo. n.d. "Técnicas de Algoritmos Básicos de Visión Artificial."
- Matteo, Leandro M Di, and Juan Carlos Gómez. 2014. "Visión Asistida Por Computador En Tiempo Real Utilizando CAMShift e Histogramas ... Algoritmo de Seguimiento de Objetos Basado En Visión Asistida Por Computador En

- Tiempo Real Utilizando CAMShift e Histogramas Ponderados,” no. November.
<https://doi.org/10.13140/2.1.2895.2641>.
- “Milk.” n.d. Accessed March 1, 2020.
http://vision.ucsd.edu/~bbabenko/new/project_miltrack.shtml.
- Muñoz, Mario A, Jesús A López, and Eduardo F Caicedo. 2008.
 “Inteligencia de Enjambres: Sociedades Para La Solución de Problemas (Una Revisión) Swarm Intelligence: Problem-Solving Societies (a Review).” *AGOSTO DE*. Vol. 28.
- RAFAEL ORMAECHEA IZQUIERDO, and .SANTIAGO AJA FERNÁNDEZ. 2015.
 “DETECCIÓN Y SEGUIMIENTO DE OBJETOS MÓVILES EN SECUENCIAS DE VIDEO.”
- Sammut, Claude, and Geoffrey I. Webb, eds. 2010. *Encyclopedia of Machine Learning. Encyclopedia of Machine Learning*. Boston, MA: Springer US.
<https://doi.org/10.1007/978-0-387-30164-8>.
- “Seguimiento de Objetos Usando OpenCV (C ++ / Python) | Aprende OpenCV.” n.d. Accessed March 14, 2020.
<https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/>.
- Wainschenker Mg Ing José María Massa Mg Ing Paula Tristan, Rubén. n.d.
 “Procesamiento Digital de Imágenes Clase Teórico Práctica N° 1 Optativa Área Procesamiento de Señales Primer Cuatrimestre de 2011.”
- Xiang, Y, A Alahi, S Savarese - Proceedings of the IEEE, and Undefined 2015. 2016. “Learning to Track: Online Multi-Object Tracking by Decision Making.” *Cv-Foundation.Org*, 9. https://www.cv-foundation.org/openaccess/content_iccv_2015/html/Xiang_Learning_to_Track_ICCV_2015_paper.html.
- Yoon, San-Ho, Bustamante Profesor, Guía : Claudio, Cubillos Figueroa, Profesor Co-Referente, and Rodrigo Alfaro Arancibia. n.d.
 “ALGORITMOS DE PROCESAMIENTO DE IMAGEN APLICADOS A LA DETECCIÓN DE FIGURAS GEOMETRICAS Y SUS PROPIEDADES ESPACIALES.”