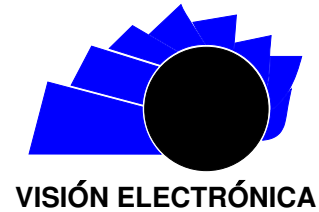




Visión Electrónica

Más que un estado sólido

<http://revistas.udistrital.edu.co/ojs/index.php/visele/index>



VISIÓN INVESTIGADORA

Detección de movimiento en tiempo real utilizando flujo óptico

Motion detection in real time using optical flow

Jonathan Avendaño P.¹, Sergio Mora M.², John E. Vera³, Javier A. Torres P.⁴

INFORMACIÓN DEL ARTÍCULO

Historia del artículo:

Enviado: Octubre de 2014

Recibido: Noviembre de 2014

Aceptado: Febrero de 2015

Palabras clave:

Detección de movimiento

Flujo óptico

SIFT

SURF



Keywords:

Motion detection

Optical flow

SIFT

SURF

RESUMEN

Se muestra la implementación de algoritmos de flujo óptico y extracción de características para el reconocimiento y detección de objetos en tiempo real de una secuencia de imágenes. Se implementaron algoritmos de flujo óptico y de detección de características SIFT, SURF y un estudio sobre la influencia de los cambios de iluminación en el ambiente de la escena analizando el funcionamiento y desempeño. Durante los cambios de iluminación se encontró que los puntos de interés entre dos imágenes consecutivas se reducen debido a que los algoritmos de detección no encuentran patrones similares y todos los niveles de intensidad son diferentes. Los resultados experimentales muestran que el algoritmo propuesto es funcional en escenas donde la luminosidad es constante y no es inferior a 1 Lux, se entrega una herramienta útil para sistemas de vigilancia y robótica móvil.

ABSTRACT

This paper shows the implementation the optical flow algorithms and feature extraction for recognition and objects detection in real time with a sequence of images. The algorithms used are optical flow, SIFT and SURF also a study on the influence of lightning changes in the environment of the scene analyzing the functioning and performance of each proposed algorithms. During lightning changes was found that points of interest between two consecutive images is reduced due to the fact that the algorithms detection are not similar patterns and its levels of intensity are different. The experimental results of research show that the proposed algorithm is functional in scenes where the brightness is constant and it is not less than 1 Lux, it provides a useful tool for surveillance systems and mobile robotics.

¹Ingeniero electrónico, Universidad ECCI, Colombia; magíster (c) Universidad Nacional de Colombia, Colombia; docente investigador, Universidad ECCI, Bogotá D.C., Colombia. Correo electrónico: javedanop@ecci.edu.co

²Ingeniero electrónico, Universidad Nacional de Colombia; magíster en Ingeniería Electrónica, Universidad de los Andes, Colombia; docente investigador, Universidad ECCI, Bogotá D.C., C Colombia. Correo electrónico: smoram@ecci.edu.co

³Ingeniero electrónico, USACA, Colombia; magíster en Ingeniería Electrónica, Pontificia Universidad Javeriana, Colombia; docente investigador, Universidad ECCI, Bogotá D.C., Colombia. Correo electrónico: jverav@ecci.edu.co

⁴Ingeniero electrónico, Universidad de San Buenaventura, Colombia; magíster (c) Pontificia Universidad Javeriana, Colombia; docente investigador, Universidad ECCI, Bogotá D.C., Colombia. Correo electrónico: atorresp@ecci.edu.co

1. Introducción

La visión por computador ha sido bastante estudiada en el área de la robótica. En una investigación sobre la oclusión consistente en el flujo óptico [1] se propuso una formulación variacional que calcula el flujo óptico y detecta las oclusiones implícitas, para luego extrapolar el flujo en dichas zonas de oclusión. Este trabajo se inspiró en el hecho de que el flujo óptico puede ser estimado con fiabilidad entre áreas visibles en dos imágenes, pero no en las zonas de oclusión. En [2], realizaron una investigación sobre la detección de movimiento fusionando el método de diferencia temporal y el campo del flujo óptico. Su trabajo se dividió en tres partes, la primera consistió en hacer la diferencia absoluta de dos imágenes consecutivas, ambas en escala de grises. La imagen resultante de la diferencia absoluta es filtrada con un filtro pasa bajos y posteriormente binarizada. El movimiento de los bordes de los objetos es indexado desde las coordenadas de las filas y las columnas. En segunda instancia se calcula el campo de flujo óptico de la secuencia de imágenes usando el algoritmo de [3]. Finalmente, el área de movimiento del objeto es encontrada fuera de los bordes indexados y arriba del campo del flujo óptico. En busca del ajuste de las regiones inestables para mejorar la calidad y la precisión del flujo óptico [4] realizaron un nuevo algoritmo que mejora la estimación del flujo óptico para detectar y reajustar regiones inestables. El estudio de la detección de movimiento ha sido extensamente estudiado, en busca de encontrar un nuevo y mejor método, la investigación de [5] juntó el análisis por flujo óptico y el análisis de componentes principales (PCA). Su enfoque mostró ser efectivo con fondos dinámicos y estáticos y particularmente útil en la detección de movimiento de escenas de exterior con baja calidad y con pequeños objetos en movimientos.

Otro modo para la detectar movimiento en escenas es por medio de la identificación de características, dos de los métodos más conocidos son Scale Invariant Features Transform (SIFT, por sus siglas en inglés) y el Speeded Up Robust Features (SURF, por sus siglas en inglés) desarrollados en 1999 y 2006 respectivamente. En [6] realizaron un estudio para la detección de sombras falsas basados en el algoritmo SIFT, desarrollaron un algoritmo que fusionó el modelo RGB de detección de regiones de sombras y la detección de características SIFT en dos cuadros de imágenes (frames) consecutivos. Este trabajo entregó un nuevo método de detección de sombras falsas que mostró ser rápido y exacto según los resultados obtenidos en su experimentación y útil en la detección de sombras para sistemas de visión de vigilancia. Un nuevo desarrollo usando SIFT fue hecho

por [7], consistió en realizar una detección de objetos usando una simple imagen de consulta sin necesidad de usar algún tipo de entrenamiento. Esta imagen de consulta puede ser real, virtual, o un simple esqueleto de un objeto. Este trabajo usó SIFT para la extracción del gradiente de distribución. Uno de los últimos avances en este tema fue realizado por [8] quienes realizaron un nuevo algoritmo llamado F-SIFT (Flip-Invariant SIFT), este método no es más que una mejora al algoritmo SIFT. Los resultados de este trabajo arrojaron superioridad del nuevo método comparado con los algoritmos de: SIFT, SURF, PCA-SIFT, RIFT, SPIN, MI-SIFT y FIND.

El algoritmo SURF es usado ampliamente en el campo de visión artificial. En [9], realizaron un trabajo para la detección de vehículos que se acercan por los lados a un vehículo en movimiento, el algoritmo de detección de características fue basado en el algoritmo SURF. Este enfoque fue probado en entornos de tráfico real obteniendo excelentes resultados. Un nuevo método para la detección de objetos en movimientos con fondos no estacionarios fue realizado por [10]. En los cambios de rotación o de escala producidos por la cámara, el algoritmo de SURF es usado para la extracción de las características invariantes. Los resultados experimentales mostraron que este algoritmo es efectivo y flexible para su propósito. Con el fin de mejorar el gasto computacional, en [11] desarrollaron un nuevo método para la detección de objetos en movimiento basados en un algoritmo mejorado del SURF. Se mejoró el gasto computacional del tradicional SURF limitando el número de características detectadas y adaptando un método rápido para reducir cálculos repetidos. El enfoque de este trabajo mostró robustez, altas precisiones y superioridad en gastos de cómputo comparado con otros métodos existentes en detección de objetos basados en SIFT y SURF.

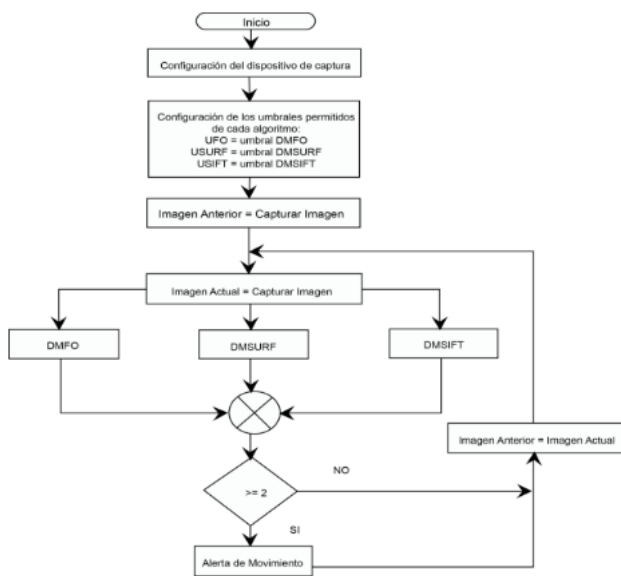
El presente trabajo busca entregar una herramienta computacional útil para robótica móvil y sistemas de vigilancia en la que se puede detectar el movimiento de objetos por medio de flujo óptico y detección de características. Para esto se hizo uso de los algoritmos de flujo óptico SURF y SIFT debido a su robustez, alta precisión y uso constante en el estudio de detección de movimiento. El método de detección de flujo óptico de [3] fue escogido para este trabajo con base en el estudio realizado por [12], en el cual se evidencia que tiene mayor precisión calculando el flujo óptico de pequeños objetos comparado con otros métodos usados para este fin. SIFT fue escogido como algoritmo para la detección de características con base al estudio de [13], donde se comparó el método SIFT con los métodos de detección de esquinas de Forstner, Harris y SUSAN. Mostrando que el algoritmo de SIFT es mejor haciendo su trabajo que

los demás algoritmos estudiados, a la hora de cambiar la escala, rotación, nivel de ruido gaussiano e iluminación de las imágenes. SURF fue otro método implementado para detección de movimiento de objetos; se escogió basándose en los trabajos relacionadas anteriormente donde se evidencia la robustez de este algoritmo.

2. Metodología

El algoritmo para la detección de movimiento en tiempo real propuesto consiste, en primera instancia, en analizar dos imágenes consecutivas de una adquisición continua. Estas dos imágenes: imagen anterior y actual, son procesadas por los algoritmos de determinación de movimiento por medio de flujo óptico (DMFO) y los algoritmos de determinación de movimiento por medio de extracción de características SURF y SIFT (DMSURF y DMSIFT). En el caso de DMFO el algoritmo emite un “1” cuando el flujo óptico entre las imágenes es mayor al umbral establecido a priori, y “0” si es la situación contraria. Para la detección de movimiento por medio de la extracción de características con los algoritmos de DMSURF y DMSIFT, cada algoritmo emite un “1” cuando el porcentaje de coincidencias entre las dos imágenes es menor al umbral establecido a priori. El algoritmo emite una alerta de movimiento cuando al menos dos de los métodos de detección de movimiento emiten un “1”. Este proceso se repite hasta que se indique lo contrario. La Figura 1 ilustra el algoritmo propuesto para la detección de movimiento.

Figura 1: Diagrama de flujo del algoritmo propuesto

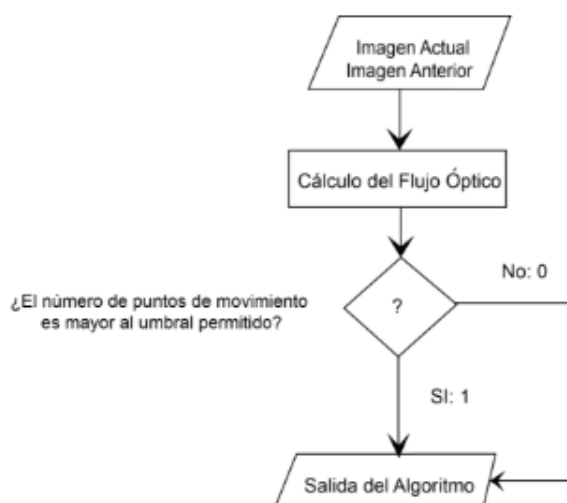


Fuente: elaboración propia.

2.1. Determinación de Movimiento por Medio de Flujo Óptico (DMFO)

El método DMFO identifica los vectores de movimiento en dos imágenes de misma escena en diferente tiempo y con el mismo tamaño, calculando el flujo óptico por medio del algoritmo dado en [3]. El flujo óptico representado en vectores, que indican la dirección y la distancia del movimiento de los píxeles, están almacenados en una matriz que es del mismo tamaño de las imágenes. El umbral permitido se establece en (%) y está relacionado con el número total de píxeles dentro de la imagen. Si el umbral se establece en 0% esto quiere decir que el mínimo cambio en cualquier píxel es tomado en cuenta para la emisión de una alarma de movimiento. Cuando el umbral es mayor que el 0% se está permitiendo el movimiento en el porcentaje establecido de píxeles dentro de la imagen, Figura 2.

Figura 2: Algoritmo DMFO



Fuente: elaboración propia.

2.2. Determinación de movimiento por medio de extracción de características SURF y SIFT (DMSURF y DMSIFT)

La detección de movimiento por medio de extracción de características analiza los puntos característicos coincidentes de dos imágenes. Las características de cada imagen se encuentran por medio de los métodos SURF y SIFT. Cada uno de estos métodos identifica dentro de cada imagen puntos característicos y luego analiza cuántos de estos puntos están presentes en ambas imágenes. Los puntos de coincidencia (keypoints) que cada algoritmo detectó (SIFT y SURF) son comparados

con el umbral definido a priori, y el algoritmo emite "1" cuando el número de coincidencias de características encontrado es menor al umbral permitido. Este umbral se establece en porcentaje y está relacionado con el número de características encontradas en la primera imagen (imagen anterior).

Si el umbral se establece en 100% esto quiere decir que si el número de coincidencias encontradas entre las dos imágenes es menor al número de características en la primera imagen se emite una alerta de movimiento. A medida que el porcentaje del umbral disminuye se permite mayor diferencia entre las coincidencias de ambas imágenes y las características halladas de la primera imagen. Los algoritmos DMSIFT y DMSURF funcionan por separado y emiten una alarma diferente. A continuación, se ilustran los algoritmos de determinación de movimiento por medio de extracción de características, Figura 3 y Figura 4.

2.3. Configuración del experimento

Para este caso, la etapa experimental fue realizada en MATLAB® 2012b; se usaron las herramientas de image acquisition toolbox e image processing toolbox. Para los algoritmos de detección de características SURF y detección de flujo óptico con el método de [3] se usaron las librerías de MATLAB®. En el caso del algoritmo de SIFT se usó la librería para MATLAB® [14], disponible en línea.

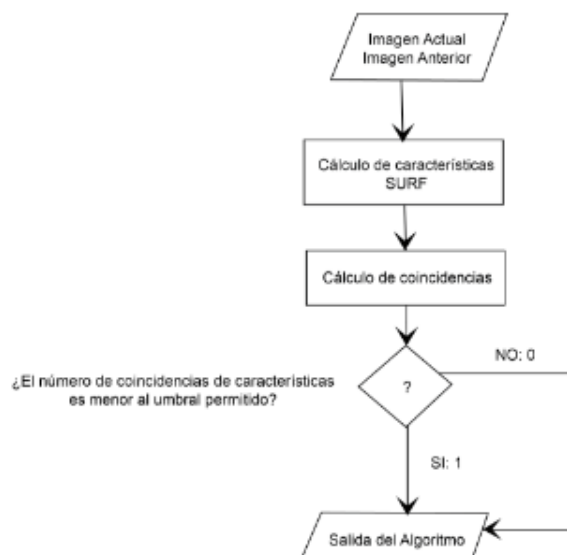
Lo primero que se debe hacer para ejecutar el algoritmo propuesto para la detección de movimiento en tiempo real, es configurar el dispositivo de captura.

Figura 1. Esto se hace por medio de la image acquisition toolbox, con la ayuda de esta es posible configurar parámetros de brillo, saturación, resolución, formato de entrada y de salida del video, forma de disparo, frames por captura, entre otros parámetros. Estos parámetros cambian dependiendo de la cámara usada, en esta investigación se usó una cámara Logitech Quick Cam Chat. Para la experimentación, se usaron los siguientes parámetros:

- Brillo: 0
- Saturación: 69
- Contraste: 17
- Resoluciones: 176x144, 320x240 y 1280x720
- Formato de entrada: RGB24
- Modo de balanceo de blanco: Automático
- Forma de disparo: Manual (Disparo inicial)
- Frames por captura: 1

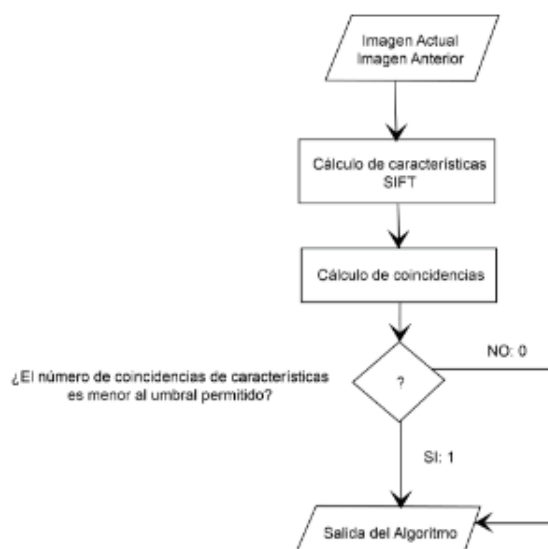
Para la facilidad de este proyecto se creó una interfaz gráfica de usuario (GUI) en MATLAB®. Los experimentos se realizaron con los siguientes niveles de umbral: UFO = 5%, USIFT = 70% y USURF = 80%.

Figura 3: Algoritmo DMSURF



Fuente: elaboración propia.

Figura 4: Algoritmo DMSIFT



Fuente: elaboración propia.

3. Resultados

Los cambios de luminosidad pueden presentarse en un entorno abierto donde una nube puede ocultar la luz solar y disminuir la cantidad de luminosidad de una escena. Por esta razón, en primera instancia, se realizaron por separado experimentos de cambios de luminosidad (lúmenes) a los algoritmos de flujo óptico, SURF y SIFT. Esto con el fin de determinar el comportamiento de cada uno de estos ante los cambios de luz. Para esto se usó un Luxómetro con el cual se determinó el número de lúmenes en cada ensayo. Ante los cambios de luz, en el caso del flujo óptico, se midió el tiempo de procesamiento; para los algoritmos de SURF y SIFT, aparte del tiempo de procesamiento, se midió el número de coincidencias encontradas. La Tabla 1 ilustra los resultados de las pruebas mencionadas.

En ella, además, se muestran los resultados de tres ensayos con diferentes escenarios donde se cambió el nivel de luminosidad y se midió el comportamiento de los algoritmos implementados para la detección del movimiento. Los parámetros de configuración en todos los escenarios fue el mismo sin embargo de acuerdo a la Tabla 1 los tiempos entre escenarios (ensayos) cambia.

Esto se presenta debido a que el número de características en cada escenario es diferente y la cantidad de estas características afectan el tiempo de

cálculo de los algoritmos. Según los resultados de la Tabla 1 también se puede evidenciar que el tiempo de cálculo de flujo óptico es proporcional al nivel de luminosidad, esto se presenta debido a que a medida que disminuye luminosidad, la imagen se vuelve negra y por ende los cambios de intensidad en los pixeles se vuelven menos perceptibles. El tiempo de ejecución de los algoritmos de extracción de características tiene un comportamiento opuesto al del flujo óptico, el tiempo de cálculo es inversamente proporcional a la luminosidad. Esto se presenta porque a menos luz, en la escena la identificación de puntos característicos, es una tarea más difícil, pues no encuentra características y, como consecuencia, el tiempo de cálculo es mayor. El número de coincidencias también decrece si la luminosidad de la escena disminuye. Otro resultado que se pudo evidenciar en esta prueba fue el siguiente, al disminuir la iluminación de la escena, el algoritmo SIFT genera coincidencias erróneas y SURF genera pocas coincidencias.

Otra prueba que se realizó, consistió en encontrar la mínima cantidad de luminosidad necesaria para que cada algoritmo funcionara. Los resultados arrojaron que la cantidad mínima de lúmenes varía dependiendo de la resolución de la imagen. Para la resolución antes mencionada en la configuración del experimento se encontró que el valor mínimo de lúmenes para el algoritmo DMFO es de 0.5 Lux y para los algoritmos de DMSIFT y DMSURF el valor mínimo es de 1 Lux.

Tabla 1: Resultados experimentación cambios de luminosidad

NUMERO DE ENSAYO	NIVEL DE ILUMINACION (x100) LUX		ALGORITMO				
	Valor Inicial	Valor Final	FLUJO OPTICO	SIFT		SURF	
			Tiempo (seg).	Tiempo (seg).	No.Keypoints	Tiempo (seg).	No.keypoints
1	1528	1509	4,815	0,625	163	0,255	80
	1432	1399	4,696	0,736	98	0,391	69
	1370	900	4,228	0,932	68	0,781	58
	800	120	4,18	950	29	0,823	45
	119	106	4,142	0,96	25	0,891	36
	106	75	4,12	0,98	20	0,915	21
2	75	106	5,196	0,378	57	0,283	79
	106	119	5,215	0,365	94	0,25	90
	120	800	5,223	0,361	102	0,243	109
	900	1370	5,234	0,352	110	0,223	112
	1399	1432	5,655	0,344	120	0,215	125
	1509	1528	5,765	0,315	145	0,201	128
3	133	131	4,181	0,375	117	0,183	35
	104	99	4,01	0,457	80	0,2	31
	91	80	3,951	0,537	59	0,234	29
	66	10	3,677	0,562	35	0,256	24
	2	0,5	3,615	0,575	16	0,278	12
	0,03	0,029	3412	0,583	10	0,312	8

Fuente: elaboración propia.

Se realizaron pruebas sobre una escena manteniendo el mismo nivel de luminosidad, pero cambiando la resolución del video, los resultados se ilustran en la Tabla 2.

Tabla 2: Resultados experimentación cambio de resolución

RESOLUCION	ALGORITMO				
	FLUJO OPTICO	SIFT		SURF	
	Tiempo (seg).	Tiempo (seg).	No.keypoints	Tiempo (seg).	No.keypoints
176x144	215	98	108	60	50
320x240	551	107	282	154	115
1280x720	6221	2394	1064	1416	429

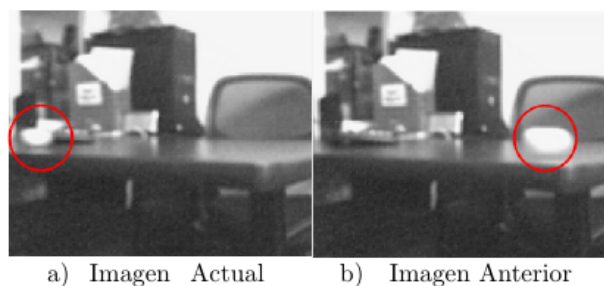
Fuente: elaboración propia.

En la Tabla 2 se evidencia que el número de características encontradas por los algoritmos de SIFT y SURF aumenta proporcionalmente con la calidad, esto es bueno y se produce debido a que existen más puntos de interés. El tiempo de cálculo es proporcional a la resolución de la imagen. Se encontró que estos tiempos varían dependiendo de los procesos que se estén ejecutando en ese momento en el computador y del tipo de computador que se esté usando.

Por último, se procedió a evaluar la herramienta creada para la detección de movimientos. Las pruebas arrojaron que el algoritmo es bueno detectando movimiento en ambientes que cumplen con la mínima cantidad de lúmenes y donde la luminosidad se mantiene. Si el algoritmo DMFO falla en entornos de baja iluminación, los algoritmos de extracción de características DMSURF y DMSIFT emiten una alerta si existe al menos un nivel de gris diferente en la imagen y este se mueve. De las pruebas realizadas con objetos de diferente tamaño y color; el algoritmo tuvo un buen funcionamiento en ambientes de luminosidad constante, incluso con objetos con tamaño pequeño como por ejemplo una pelota de pin-pong. El tiempo de cálculo de todo el algoritmo fue aproximadamente de un segundo, esto limitó la capacidad de detectar objetos que se movieran a una velocidad mayor a esta. El algoritmo es muy sensible a los cambios de luz, si en una escena el nivel de luz cambia el método puede generar falsas alarmas.

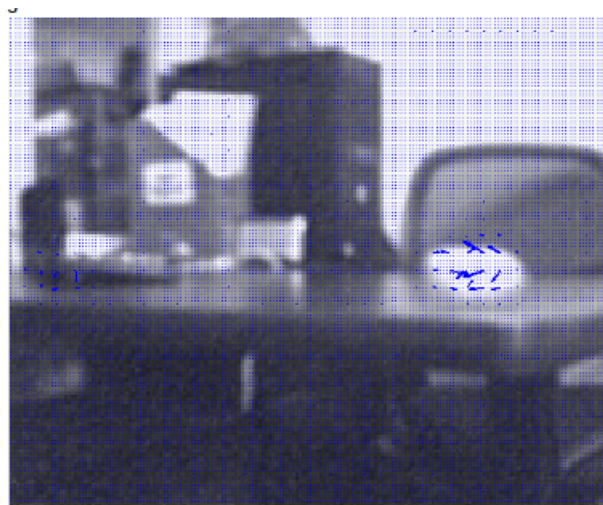
En la Figura 5 se muestra una escena de prueba en la que se lanzó una pelota de ping-pong la cual está encerrada en el círculo resaltado en ambas imágenes. A continuación, se muestra la salida de cada uno de los algoritmos propuestos para la detección de movimiento, Figura 6.

Figura 5: Escena de Prueba



Fuente: elaboración propia.

Figura 6: Vectores detectados por el algoritmo de Flujo Óptico

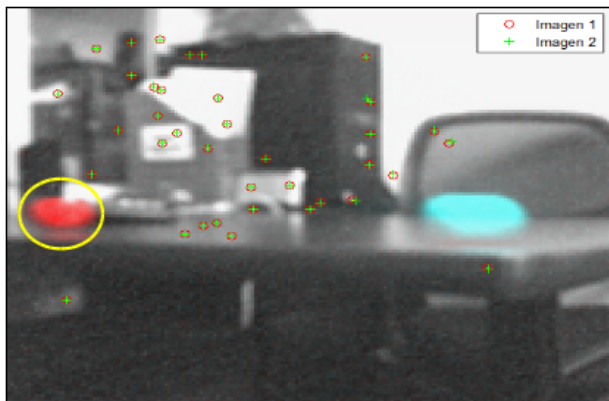


Fuente: elaboración propia.

En la Figura 7 el círculo resaltado encierra una zona de color dentro de la imagen, esta zona representa un

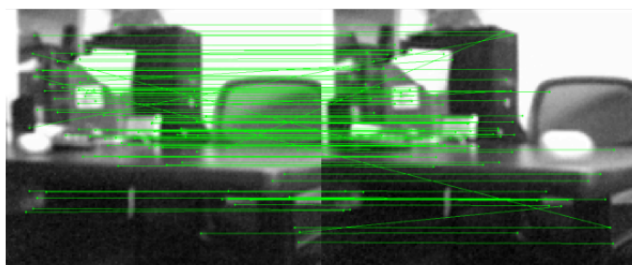
objeto que en la primera imagen estaba en este lugar y que el algoritmo reconoció como característica pero que en la segunda imagen no se encontraba en la misma posición, Figura 8.

Figura 7: Coincidencias detectadas por el algoritmo SURF



Fuente: elaboración propia.

Figura 8: Coincidencias detectadas por el algoritmo SIFT



Fuente: elaboración propia.

4. Conclusiones

En este proyecto se presentó un método para la detección de movimiento que integró el algoritmo de flujo óptico y los algoritmos de extracción de características SIFT y SURF. En los experimentos realizados sobre los algoritmos de flujo óptico, SIFT y SURF, donde se varió la luminosidad con el fin de evaluar el comportamiento de cada algoritmo, se encontró que durante los cambios de iluminación los puntos de coincidencias entre dos imágenes consecutivas de una escena se reducen debido a que los algoritmos de extracción de características no encuentran patrones similares y todos los niveles de intensidad son diferentes.

En el caso de SIFT se generan coincidencias erróneas entre puntos de características encontrados, y SURF genera pocas coincidencias. Los tiempos de ejecución de los algoritmos SIFT y SURF son inversamente proporcionales a la iluminación. Para el caso del Flujo Óptico se encontró una relación proporcional del tiempo de ejecución y la iluminación. Se obtuvo también que el comportamiento del algoritmo depende de la resolución del video: si la resolución es muy alta se puede monitorear más puntos de características y por ende tener más precisión pero con un coste de tiempo de ejecución más alto. Se encontró que el método propuesto para la detección de movimiento en tiempo real funciona muy bien en ambientes donde la luminosidad es constante pero no es menor que los límites encontrados (1 Lux), entregando un buen método para sistemas de vigilancia y de robótica móvil, debido a que es capaz de detectar movimiento de objetos tan pequeños como el de una pelota de ping-pong.

Referencias

- [1] S. Ince, and J. Konrad, "Occlusion-Aware Optical Flow Estimation". *Image Processing, IEEE Transactions on*, vol.17, no.8, pp.1443-1451, aug, 2008, doi: 10.1109/TIP.2008.925381
- [2] W. Shuigen, C. Zhen, and D. Hua, "Motion Detection Based on Temporal Difference Method and Optical Flow field". *Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium on*, vol.2, pp.85-88, 24 may, 2009, doi: 10.1109/ISECS.2009.62
- [3] K.P. Berthold H., and B.G. Schunck, "Determining Optical Flow". *Technical Report. Massachusetts Institute of Technology, Cambridge, MA, USA, 1980.*
- [4] S. Viet-Uyen H. and J. Wook J., "Readjusting Unstable Regions to Improve the Quality of High Accuracy Optical Flow". *Circuits and Systems for Video Technology, IEEE Transactions on*, vol.20, no.4, pp.540-547, april, 2010, doi: 10.1109/TCSVT.2010.2041818
- [5] K. Liu, H. Yang, B. Ma, and Q. Du, "A joint optical flow and principal component analysis approach for motion detection". *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp.1178-1181, march, 2010, doi: 10.1109/ICASSP.2010.5495379
- [6] L. Zhang and X. He, "Fake Shadow Detection Based on SIFT Features Matching". *Information Engineering (ICIE), 2010 WASE International*

- Conference on, vol.1, pp.216-220, 14-15, aug, 2010, doi: 10.1109/ICIE.2010.58
- [7] B. Xiong, and D. Xiaoqing, "A generic object detection using a single query image without training". *Tsinghua Science and Technology*, vol.17, no.2, pp.194-201, april, 2012, doi: 10.1109/TST.2012.6180045
- [8] W.-L. Zhao and C.-W. Ngo, "Flip-Invariant SIFT for Copy and Object Detection". *Image Processing, IEEE Transactions on*, vol.22, no.3, pp.980-991, march, 2013, doi: 10.1109/TIP.2012.2226043
- [9] W.-C. Chang and K.-J. Hsu, "Vision-based side vehicle detection from a moving vehicle". *System Science and Engineering (ICSSE), 2010 International Conference on*, pp.553-558, july, 2010, doi: 10.1109/ICSSE.2010.5551779
- [10] Y. Chen *et. al.* "Object detection and tracking with active camera on motion vectors of feature points and particle filter". *Review of Scientific Instruments*, vol 84, no 6, 2013.
- [11] J. Pan, W. Chen and W. Peng, "A new moving objects detection method based on improved SURF algorithm". *Control and Decision Conference (CCDC), 2013 25th Chinese*, pp.901-906, may, 2013, doi: 10.1109/CCDC.2013.6561051
- [12] A. Bruhn, J. Weickert, and C. Schnörr, "Combining local and global optic flow methods". *International Journal of Computer Vision*, pp. 211-231, 2205.
- [13] D. Jiang, and J. Yi, "Comparison and Study of Classic Feature Point Detection Algorithm". *Computer Science & Service System (CSSS), 2012 International Conference on*, pp.2307-2309, pp. 11-13, aug, 2012 doi: 10.1109/CSSS.2012.572
- [14] A. Vedaldi, "An open implementation of the SIFT detector and descriptor". *An open implementation of the SIFT detector and descriptor*, 2007.