

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

**Implementación y comparación de los algoritmos de
determinación de flujo óptico de Horn-Schunck y Lucas-
Kanade para rastreo de objetos.**

Estudios Comparados

Ricardo Esteban Pazmiño Reyes

Ingeniería Eléctrica y Electrónica

Trabajo de titulación presentado como requisito
para la obtención del título de
Ingeniero Electrónico

Quito, 02 de abril de 2016

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ
COLEGIO POLITECNICO

**HOJA DE CALIFICACIÓN
DE TRABAJO DE TITULACIÓN**

**Implementación y comparación de los algoritmos de determinación de flujo
óptico de Horn-Schunck y Lucas-Kanade para rastreo de objetos.**

Ricardo Esteban Pazmiño Reyes

Calificación:

Nombre del profesor, Título académico

Luis Miguel Procel , Ph.D.

Firma del profesor

Quito, 02 de abril de 2016

Derechos de Autor

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Firma del estudiante: _____

Nombres y apellidos: Ricardo Esteban Pazmiño Reyes

Código: 00109000

Cédula de Identidad: 1715829576

Lugar y fecha: Quito, 30 de abril del 2016

RESUMEN

Este trabajo presenta la implementación y una breve comparación del algoritmo para la determinación de flujo óptico de Berthold Horn y Brian Schunck y el algoritmo desarrollado con el mismo fin por Bruce Lucas y Takeo Kanade, utilizando la plataforma de Matlab. El método implementado por Horn y Schunck toma dos restricciones para su correcto funcionamiento en una secuencia de imágenes consecutivas (video): el cambio de brillo entre dos cuadros consecutivos del video es cero o aproximadamente cero (se asume constancia de brillo), y, la minimización de las primeras derivadas del flujo (restricción de suavidad). El método desarrollado por Lucas y Kanade asume la misma restricción de constancia de brillo que el método de Horn y Schunck pero difiere de éste por la segunda restricción, asumiendo un vector de velocidades constante dentro de un vecindario de dimensiones pequeñas. Para comprender el comportamiento de ambos algoritmos se varían los parámetros característicos de cada uno y se concluye sobre los cambios en el flujo óptico. Con la finalidad de analizar cuán óptimo es el primer método respecto al segundo los algoritmos son comparados en tiempo y estructura.

Palabras clave: Horn & Schunck, Lucas Kanade, flujo óptico, video, cuadro, rastreo de objetos, visión por computadora, constancia de brillo, restricción de suavidad, vecindarios.

ABSTRACT

This paper presents the implementation and a brief comparison of the algorithm for determining the optical flow developed by Berthold Horn and Brian Schunck and the algorithm developed for the same purpose by Bruce Lucas and Takeo Kanade, using the Matlab platform. The method proposed by Horn & Schunck takes as constraints two assumptions for its correct functioning on a sequence of consecutive images (video): the change of the brightness between two consecutive frames of the video is zero or approximates to zero (brightness constancy assumption), and the first derivatives of the optical flow are minimized (smoothness constraint). The method developed by Lucas & Kanade uses the same brightness constancy assumption as Horn & Schunck but differs from this one on the second constraint, assuming a constant velocity vector inside a small neighborhood. In order to understand the behavior of both algorithms, certain characteristic parameters are varied and it is concluded about the changes on the optical flow. With the aim of analyzing how optimal is one method over the other one, the algorithms are compared on time and structure.

Key words: Horn & Schunck, Lucas Kanade, optical flow, video, frame, tracking, computer vision, brightness constancy, smoothness constraint, neighborhoods.

TABLA DE CONTENIDO

Introducción.....	10
Visión por Computadora.....	10
Movimiento.....	11
Estimación de Derivadas por Diferencias Finitas y Detección de Bordes.....	12
Flujo Óptico.....	14
Desarrollo del Tema	18
Planteamiento del Problema	18
Restricción de Constancia de Brillo	18
Restricción de Suavidad	21
Restricción de Gradiente	23
Algoritmo de Horn y Schunck	23
Algoritmo de Lucas y Kanade.....	25
Resultados.....	26
Resultados del algoritmo de Horn & Schunck.....	28
Resultados del algoritmo de Lucas y Kanade.....	34
Análisis de Resultados.....	40
Conclusiones	43
Referencias bibliográficas.....	45

ÍNDICE DE TABLAS

Tabla 1: Parámetros variados y resultados de algoritmo implementado de Horn y Schunck. 28

Tabla 2: Parámetros variados y resultados de algoritmo implementado de Lucas y Kanade. 35

ÍNDICE DE FIGURAS

- Figura 1:** En **a.** se presenta una imagen en escala de grises de una parte de una Cebra. La derivada parcial en dirección y se presenta en **b.** (responde fuertemente a rayas horizontales y débilmente a rayas verticales) mientras que en **c.** se presenta la derivada parcial en la dirección x (que responde de manera inversa a la anterior) (**Forsyth & Ponce, 2012**)..... 13
- Figura 2:** Se muestra en **a.** una secuencia de imágenes que forma parte del video “Hamburg Taxi” del cual se han hecho varias pruebas para determinar el flujo óptico. En **b.** se observa el flujo óptico resultante de la secuencia “Hamburg Taxi” (**Shah M. , 2011**). ... 15
- Figura 3:** En la imagen mostrada en **a.** se muestra un cuadro de un video en el que la cámara cambia de posición. El flujo óptico obtenido (una ventana que enfoca al árbol) sin utilizar un método con implementación de pirámide de imágenes se muestra en **b.**, mientras que al utilizar este método se obtiene un mejor campo vectorial, como se muestra en **c.** (**Shah M. , 2011**). 16
- Figura 4:** Cubo para ponderación de valores de brillo utilizado para la aproximación de las derivadas parciales (**Horn & Schunck, 1980**)..... 20
- Figura 5:** Línea de restricción del flujo óptico comprendida en el plano formado por las velocidades u y v . El vector perpendicular a la línea de restricción muestra el gradiente de brillo E_x, E_y (**Horn & Schunck, 1980**)..... 20
- Figura 6:** Cuadrado de ponderaciones con los pesos correspondientes a cada vecino para el cálculo del Laplaciano en las componentes de u y v del flujo óptico (**Horn & Schunck, 1980**)..... 22
- Figura 7:** Dos imágenes consecutivas de la secuencia de video “RubberWhale”, editada en Matlab para tener una malla de referencia..... 27

Figura 8: Resultado para Horn y Schunck con $Nit = 100$ y $\alpha = 100$	29
Figura 9: Resultado para Horn y Schunck con $Nit = 100$ y $\alpha = 50$	30
Figura 10: Resultado para Horn y Schunck con $Nit = 25$ y $\alpha = 200$	31
Figura 11: Resultado para Horn y Schunck con $Nit = 25$ y $\alpha = 2$	32
Figura 12: Resultado para Horn y Schunck con $Nit = 2$ y $\alpha = 200$	33
Figura 13: Resultado para Horn y Schunck con $Nit = 2$ y $\alpha = 2$	34
Figura 14: Resultado para Lucas y Kanade con $Pasos = 2$	35
Figura 15: Resultado para Lucas y Kanade con $Pasos = 8$	36
Figura 16: Resultado para Lucas y Kanade con $Pasos = 32$	37
Figura 17: Resultado para Lucas y Kanade con $Pasos = 64$	38
Figura 18: Resultado para Lucas y Kanade con $Pasos = 80$	39
Figura 19: Resultado para Lucas y Kanade con $Pasos = 100$	40
Figura 20: Imágenes de muestra para comprensión de diferencias conforme a cambios en dimensiones de vecindarios.....	42

INTRODUCCIÓN

Los avances tecnológicos en los campos de la informática y la electrónica han permitido el desarrollo de hardware y software con la finalidad de permitir a una máquina el simular la comprensión (en determinado nivel) de ciertos fenómenos físicos que generalmente pasan inadvertidos por los seres vivos debido a su propia naturaleza. El desarrollo en el campo de la ciencia computacional ha dado paso a la optimización de algoritmos que permiten a las máquinas interactuar y recopilar información del mundo real de manera cada vez más eficiente, aminorando el tiempo de cálculo de procesos matemáticos esenciales para ese fin.

Visión por Computadora

La visión por computadora es una rama de la ciencia computacional que busca comprender de manera matemática los procesos biológicos de la interacción ojo-cerebro con nuestro mundo tridimensional y plasmar estos estímulos en un modelo sistemático comprendido por un dispositivo electrónico para su posterior procesamiento y uso. Sin embargo, los avances realizados en este campo son limitados y su desarrollo está aún en proceso. Lo que busca esta rama es describir a raíz de una o varias imágenes el mundo que observamos y reconstruir sus propiedades como lo son las formas, los colores, las sombras, entre otros (Szeliski, 2010).

El campo de la visión puede tornarse realmente complejo al tratar de simular sus procesos de manera matemática debido a que cuando el ser humano usa el sentido de la vista para comprender su entorno, el cerebro capta información a través de los ojos y realiza ciertos supuestos del entorno para poder interpretar parámetros como la profundidad, la reflexión de brillo sobre una superficie y el movimiento relativo de un objeto, todo esto con la finalidad de

realizar modelos en 3D que plasmen lo que está ocurriendo en el mundo real. Estos supuestos que ayudan a la comprensión de nuestro entorno se pueden convertir en un problema al momento de la elaboración de algoritmos de visión por computadora debido a que éstos deben ser basados en modelos físicos, matemáticos y/o probabilísticos para encontrar soluciones potenciales que capten de manera eficiente la cotidianidad (Szeliski, 2010).

A pesar de la complejidad de desarrollo de los algoritmos de visión por computadora, los avances han hecho posible su pronta utilización en varios campos. Entre las aplicaciones que tiene la visión por computadora en la industria y en la investigación están: reconocimiento óptico de caracteres (OCR por sus siglas en inglés), inspección de máquinas, modelado en 3D, imágenes médicas, monitoreo de tráfico, seguridad automotriz, reconocimiento de huellas entre otras (Szeliski, 2010).

Movimiento

Uno de los factores más críticos e importantes para todo ser viviente al interactuar con su entorno es la percepción y comprensión de cambio en el mismo (Mallot & Allen, 2000). La interpretación de cambio de movimiento en una secuencia de imágenes es particularmente trascendental en las aplicaciones de visión por computadora y procesamiento de video, siendo utilizada ampliamente en campos como la compresión de videos tales como MPEG (Szeliski, 2010). Por ende, algoritmos para la comprensión del movimiento de un objeto por medio de una cámara de video han venido desarrollándose desde hace ya algunos años.

El movimiento en el contexto de imágenes puede ser malinterpretado al momento de tratar algoritmos matemáticos debido a que éste puede ser confundido con factores externos como el cambio de la posición de la fuente de luz o una regulación de la misma (atenuación o incremento de la intensidad de la luminosidad del foco). A diferencia de estos casos, un

movimiento “real” muestra un desplazamiento coherente de objetos o partes de la imagen en un video (Mallot & Allen, 2000).

La facilidad de análisis de movimiento en un video dependerá directamente de la presencia de efectos externos al desplazamiento como son la rotación y el escalamiento de los objetos en los *cuadros*¹. De existir este tipo de efectos el cálculo se torna especialmente difícil. Al contrario, si se trata netamente de un desplazamiento espacial, habrá que realizar una evaluación simultánea de los cuadros de enfoque tratándolos tanto espacial como temporalmente (Mallot & Allen, 2000).

Estimación de Derivadas por Diferencias Finitas y Detección de Bordos

Los cambios bruscos de la intensidad de brillo en una imagen son de gran utilidad en el análisis de imágenes y su procesamiento. Los algoritmos que detectan estos cambios son conocidos como algoritmos de detección de bordos debido a que las fronteras de los objetos y el contraste con el fondo de la imagen o con un objeto que se encuentra ubicado detrás del primero generan cambios abruptos de brillo (Forsyth & Ponce, 2012). Uno de estos cambios bruscos (borde) se describe como un gradiente de brillo grande en magnitud en uno o varios segmentos de la imagen analizada.

Para poder calcular los gradientes en una imagen discreta (conjunto de píxeles) se requieren métodos para realizar aproximaciones de las derivadas, uno de los cuales es por diferencias finitas. Esto se explica al centrarnos en la definición de una derivada en función de sus límites, como se muestra en la Ecuación 1 (Forsyth & Ponce, 2012).

$$\frac{\partial f}{\partial x} = \lim_{\epsilon \rightarrow 0} \frac{f(x+\epsilon, y) - f(x, y)}{\epsilon} \quad (\text{Ec. 1})$$

¹ Un *cuadro* o *frame* es una imagen que forma parte de un video.

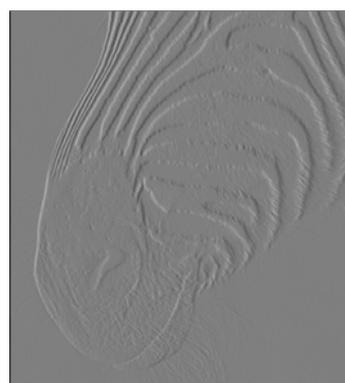
Las derivadas calculadas por diferencias finitas responden diferente al calcularse en una u otra dirección, como se muestra en la Figura 1. En una secuencia de imágenes se debe considerar (aparte de las derivadas espaciales) las derivadas temporales, debido a que se trata de una variación en espacio y tiempo a la vez.



a. Imagen de Cebra en escala de grises



b. Derivada parcial respecto a y



c. Derivada parcial respecto a x

Figura 1: En **a.** se presenta una imagen en escala de grises de una parte de una Cebra. La derivada parcial en dirección y se presenta en **b.** (responde fuertemente a rayas horizontales y débilmente a rayas verticales) mientras que en **c.** se presenta la derivada parcial en la dirección x (que responde de manera inversa a la anterior) (Forsyth & Ponce, 2012).

Para la determinación del flujo óptico se requiere adicionalmente el cálculo del Laplaciano (segunda derivada) de los vectores de flujo. De igual manera que para las derivadas espaciales y temporales, el Laplaciano se puede aproximar utilizando diferencias

finitas. Se muestra en la Ecuación 2 el concepto del Laplaciano para una función en 2D (Forsyth & Ponce, 2012).

$$(\nabla^2 f)(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (\text{Ec. 2})$$

Flujo Óptico

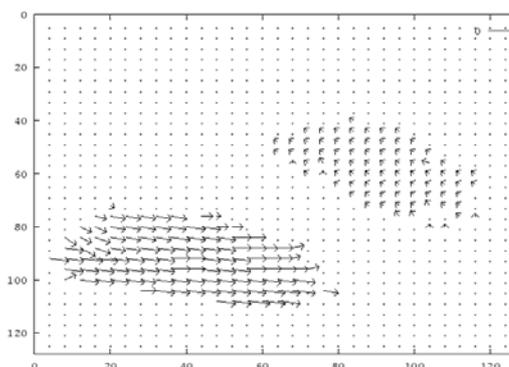
El movimiento de objetos en el mundo real (3D) induce un movimiento en 2D en el plano de las imágenes (Shah M. , 1997). Este movimiento inducido en 2D puede ser descrito a su vez como un cambio en los patrones de brillo observados, y se lo puede divisar al analizar dos cuadros consecutivos de un video. El flujo óptico describe la distribución de las velocidades aparentes en este cambio de brillo en una secuencia de imágenes (Horn & Schunck, 1980). A estos vectores de velocidad se les denomina aparentes debido a que el movimiento detectado puede surgir por el movimiento del objeto como tal o por el movimiento de la cámara estando el objeto en análisis estático. En el caso de no existir una variación apreciable en la fuente de luz, el cambio de posición de un determinado patrón de brillo entre dos o más cuadros se puede reconocer como el movimiento del objeto en análisis en la secuencia de imágenes en cuestión. Este concepto es comprensible si se tiene en consideración la definición de movimiento en video así como la detección de bordes y, como se describirá en la sección *Desarrollo del Tema*, se toman restricciones para el análisis del movimiento de los patrones de brillo.

La determinación del flujo óptico es de mucha trascendencia para el análisis de video debido a que los vectores de velocidad calculados dan información respecto al cambio de la disposición espacial de uno o varios objetos en análisis (Horn & Schunck, 1980). Al conocer el campo vectorial dado por el flujo óptico se pueden encontrar regiones con discontinuidades y, de esta manera, segmentar la imagen obteniendo delimitantes entre objetos (Horn & Schunck, 1980). Un ejemplo se muestra en la Figura 2 donde se pueden distinguir y

diferenciar dos automóviles de su fondo y entre sí por medio de la determinación del flujo óptico.



a. Hamburg Taxi



b. Flujo óptico de secuencia “Hamburg Taxi”

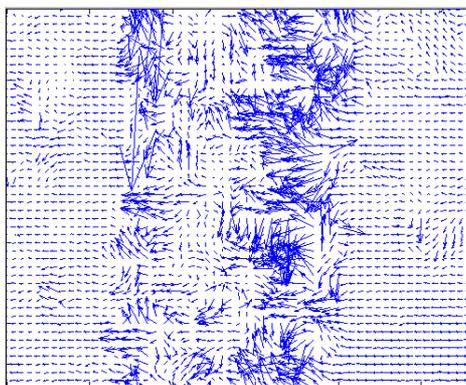
Figura 2: Se muestra en **a.** una secuencia de imágenes que forma parte del video “Hamburg Taxi” del cual se han hecho varias pruebas para determinar el flujo óptico. En **b.** se observa el flujo óptico resultante de la secuencia “Hamburg Taxi” (Shah M. , 2011).

El proceso para estimar el movimiento en una secuencia de imágenes consecutivas (video) consiste en primer lugar en escoger una métrica de error que permita comparar las imágenes de una manera adecuada. Posteriormente, se debe escoger una técnica de búsqueda de patrones similares que mejor convenga, para lo cual existen técnicas relativamente sencillas (como las que analizan de forma exhaustiva todos los alineamientos posibles entre las imágenes comparadas), pero que a su vez tienen un alto costo computacional, volviéndose

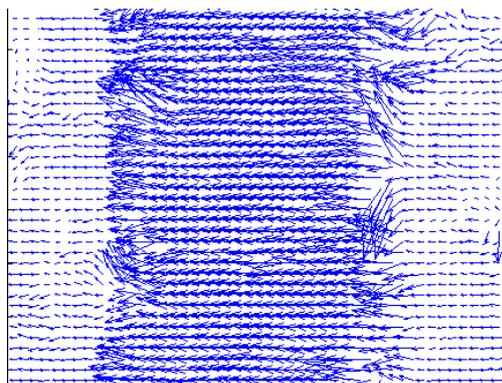
métodos poco eficientes. Por lo tanto, técnicas *grueso a fino* basadas en pirámides de imágenes junto con métodos que utilizan transformadas de Fourier para su análisis son mayormente utilizados (Szeliski, 2010). Se muestra en la Figura 3 un ejemplo de la diferencia de flujo óptico utilizando métodos de pirámides de imágenes.



a. Imagen de árbol con paisaje de fondo



b. Flujo Óptico sin pirámides



c. Flujo Óptico con pirámides

Figura 3: En la imagen mostrada en **a.** se muestra un cuadro de un video en el que la cámara cambia de posición. El flujo óptico obtenido (una ventana que enfoca al árbol) sin utilizar un método con implementación de pirámide de imágenes se muestra en **b.**, mientras que al utilizar este método se obtiene un mejor campo vectorial, como se muestra en **c.** (Shah M. , 2011).

Uno de los primeros métodos desarrollados para el cálculo de flujo óptico fue planteado por Horn y Schunck en el cual se consideran dos restricciones para su cálculo. La primera restricción planteada es la existencia de una variación mínima en los patrones de brillo entre dos cuadros consecutivos del video, también conocida como constancia de brillo.

La segunda restricción planteada es la de suavidad del campo vectorial del flujo óptico (Horn & Schunck, 1980). Posteriormente se desarrolló un algoritmo para el cálculo del flujo óptico por parte de Lucas y Kanade, quienes consideraron la misma restricción de constancia de brillo que Horn y Schunck, pero difieren en la segunda restricción planteada asumiendo que para un vecindario pequeño la variación de los vectores de velocidad va a ser muy pequeña, o en otras palabras, se asume una constancia en las componentes de la velocidad (Lucas & Take, 1981).

El flujo óptico ha tenido una gran importancia en el procesamiento de video, siendo los algoritmos desarrollados e implementados hasta la actualidad ampliamente utilizados en aplicaciones como el resumen de video, la estabilización de video, la compresión de video, y desarrollos más complejas como los utilizados para imágenes médicas y teledetección (Szeliski, 2010).

DESARROLLO DEL TEMA

Planteamiento del Problema

En la actualidad existen varios métodos de *tracking* de objetos, tanto en video como en tiempo real por medio de una cámara. Muchos de esos métodos se basan en el cálculo del flujo óptico. Estos métodos han venido siendo optimizados cada vez más, hasta obtenerse algoritmos robustos y con un costo computacional bajo. Con el fin de analizar la eficiencia de los métodos planteados, este trabajo busca implementar y comparar dos métodos desarrollados para la determinación de flujo óptico en un video: el algoritmo iterativo de Horn y Schunck versus el algoritmo de minimización de derivadas parciales de Lucas y Kanade.

La finalidad de esta comparación es determinar qué algoritmo es más eficiente en tiempo y estructura y observar la variación en los resultados de ambos algoritmos tras el cambio de parámetros característicos de cada uno de ellos.

Restricción de Constancia de Brillo

El cálculo del flujo óptico para ambos métodos comparados parte de la suposición de que el brillo se mantendrá constante al analizar dos cuadros consecutivos en el video, como se describe en la ecuación 3 (Shah M. , 1997).

$$E(x, y, t) = E(x + dx, y + dy, t + dt) \quad (\text{Ec. 3})$$

La ecuación 3 presenta una función en 3D, $E(x, y, t)$, donde E representa el brillo en un punto de la imagen, x e y representan las coordenadas espaciales de la imagen y t es una coordenada temporal en la secuencia de video. Por medio de una expansión de Taylor en la ecuación 3 se puede encontrar la ecuación 4 que describe la condición de constancia de brillo en el flujo óptico, donde E_x y E_y son las derivadas parciales de la intensidad de brillo de la

imagen respecto a la posición, E_t es la derivada parcial de la intensidad respecto al tiempo, y u , v , son las velocidades de movimiento (flujo óptico) de los píxeles en el eje x e y respectivamente (Shah M. , 1997) (Horn & Schunck, 1980).

$$E_x u + E_y v + E_t = 0 \quad (\text{Ec. 4})$$

Para el cálculo de las derivadas parciales espaciales así como para la derivada temporal se utiliza el método de diferencias finitas. Este método consiste en aproximar la derivada considerando el peso de un conjunto discreto de mediciones de brillo disponibles alrededor del píxel $E(x,y,t)$. Para la implementación de ambos algoritmos, las aproximaciones de las derivadas fueron calculadas por medio de las ecuaciones 5, 6 y 7 (Horn & Schunck, 1980):

$$E_x \approx \frac{1}{4} \{E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} + E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1}\} \quad (\text{Ec. 5})$$

$$E_y \approx \frac{1}{4} \{E_{i+1,j,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i,j+1,k} + E_{i+1,j,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i,j+1,k+1}\} \quad (\text{Ec. 6})$$

$$E_t \approx \frac{1}{4} \{E_{i,j,k+1} - E_{i,j,k} + E_{i+1,j,k+1} - E_{i+1,j,k} + E_{i,j+1,k+1} - E_{i,j+1,k} + E_{i+1,j+1,k+1} - E_{i+1,j+1,k}\} \quad (\text{Ec. 7})$$

Las aproximaciones de las derivadas propuestas en las ecuaciones 5, 6 y 7 describen la ponderación de un conjunto de valores alrededor de un punto comprendido en un cubo formado por 8 mediciones de brillo, como se muestra en la Figura 4 (Horn & Schunck, 1980).

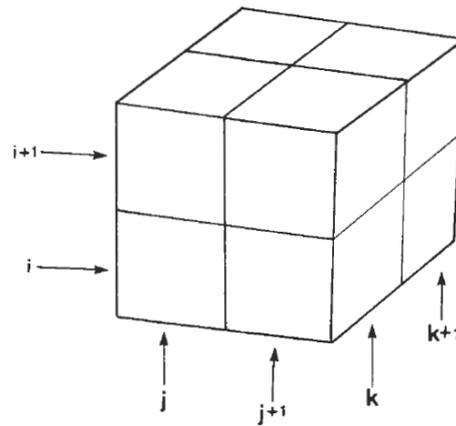


Figura 4: Cubo para ponderación de valores de brillo utilizado para la aproximación de las derivadas parciales (Horn & Schunck, 1980).

Una vez obtenidas las aproximaciones de las derivadas y , al poner una de las velocidades en función de otra en la ecuación 4, se observa que lo que se obtiene es la ecuación de una recta dentro de la cual se encuentra la solución para la determinación del flujo óptico. Se muestra en la Figura 5 la recta correspondiente a la ecuación $v = -\frac{E_x}{E_y}u - \frac{E_t}{E_y}$ (Shah M. , 2011) (Horn & Schunck, 1980).

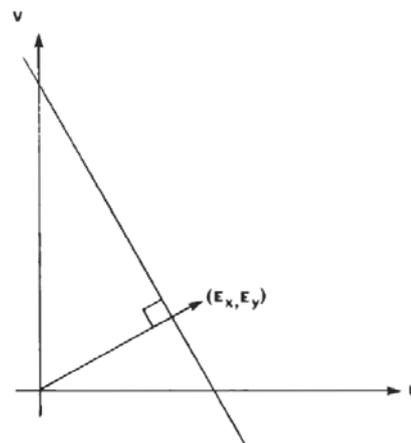


Figura 5: Línea de restricción del flujo óptico comprendida en el plano formado por las velocidades u y v . El vector perpendicular a la línea de restricción muestra el gradiente de brillo E_x, E_y (Horn & Schunck, 1980).

Como se observa en la Figura 5, a pesar de disponer de la ecuación que describe la restricción de constancia de brillo, el flujo óptico puede encontrarse a lo largo de la recta descrita en el plano $u-v$. En este punto es donde se encuentra la mayor diferencia entre los algoritmos planteados por Horn & Schunck y Lucas & Kanade, puesto a que cada método de resolución para la determinación del flujo define una segunda restricción diferente, las cuales serán descritas posteriormente.

Restricción de Suavidad

La restricción de suavidad es la segunda restricción utilizada por Horn y Schunck para la determinación del flujo óptico. Ésta asume que los pixeles que conforman los objetos de tamaño finito en la imagen (patrones de brillo) tienden a someterse a movimientos rígidos como un todo, por lo que casi nunca se encuentran pixeles con movimientos independientes de sus vecinos cercanos. Esto genera un campo de velocidades de los patrones de brillo que varía suavemente en casi toda la imagen (debido a determinadas excepciones como en el caso de presencia de texturas) (Horn & Schunck, 1980).

Esta restricción implica la minimización de las derivadas parciales espaciales de las componentes de la velocidad. Para poder realizar esta minimización se puede plantear la suma de los Laplacianos en los ejes x e y como se muestra en las ecuaciones 8 y 9 (Horn & Schunck, 1980). En estas ecuaciones se muestra una equivalencia donde se consideran promedios locales de las velocidades (\bar{u} y \bar{v}) así como un factor proporcional k (Horn & Schunck, 1980). Para lograr la minimización de los Laplacianos de las componentes de la velocidad, éstos deben ser igualados a 0.

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \approx k(\bar{u}_{i,j,k} - u_{i,j,k}) \quad (\text{Ec. 8})$$

$$\nabla^2 v = \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \approx k(\bar{v}_{i,j,k} - v_{i,j,k}) \quad (\text{Ec. 9})$$

Al igual que se utilizaron diferencias finitas para la aproximación de las derivadas parciales, el cálculo del Laplaciano utiliza el mismo método de aproximación, tomando las ponderaciones de un cuadrado de valores de los vecinos cercanos y sustrayéndolo del valor central, como se muestra en la Figura 6 (Horn & Schunck, 1980).

$\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{12}$
$\frac{1}{6}$	-1	$\frac{1}{6}$
$\frac{1}{12}$	$\frac{1}{6}$	$\frac{1}{12}$

Figura 6: Cuadrado de ponderaciones con los pesos correspondientes a cada vecino para el cálculo del Laplaciano en las componentes de u y v del flujo óptico (Horn & Schunck, 1980).

El cálculo de los promedios locales de las componentes de la velocidad se lo realiza utilizando las ecuaciones 10 y 11, donde se realiza la ponderación planteada en la Figura 6 (Horn & Schunck, 1980).

$$\begin{aligned} \bar{u}_{i,j,k} = & \frac{1}{6} \{u_{i-1,j,k} + u_{i,j+1,k} + u_{i+1,j,k} + u_{i,j-1,k}\} \\ & + \frac{1}{12} \{u_{i-1,j-1,k} + u_{i-1,j+1,k} + u_{i+1,j+1,k} + u_{i+1,j-1,k}\} \quad (\text{Ec. 10}) \end{aligned}$$

$$\begin{aligned} \bar{v}_{i,j,k} = & \frac{1}{6} \{v_{i-1,j,k} + v_{i,j+1,k} + v_{i+1,j,k} + v_{i,j-1,k}\} \\ & + \frac{1}{12} \{v_{i-1,j-1,k} + v_{i-1,j+1,k} + v_{i+1,j+1,k} + v_{i+1,j-1,k}\} \quad (\text{Ec. 11}) \end{aligned}$$

La restricción de suavidad es una suposición bastante buena para el cálculo de flujo óptico excepto cuando existen objetos en la imagen que ocultan a otros, provocando de esta manera una discontinuidad en el flujo (Horn & Schunck, 1980).

Restricción de Gradiente

El método desarrollado por Lucas y Kanade plantea como segunda restricción que el vector de velocidades es constante (constancia del gradiente de brillo) en un vecindario relativamente pequeño en comparación al tamaño de la imagen en cuestión (Shah M. , 2011). Para poder comprender de manera matemática esta consigna, se utiliza una función dependiente de los vectores de velocidad, como se muestra en la ecuación 12 (Bagci, 2015).

$$f(\mathbf{u}, \mathbf{v}) = \iint (E_x \mathbf{u} + E_y \mathbf{v} + E_t)^2 dx dy \quad (\text{Ec. 12}).$$

Con la finalidad de obtener coherencia en el movimiento de los patrones de brillo a lo largo de la imagen, se optimiza la Ec. 12 por medio de las derivadas parciales de ésta con respecto a la velocidad e igualando este gradiente a cero, como se muestra en la ecuación 13 (Bagci, 2015).

$$\frac{\partial f(\mathbf{u}, \mathbf{v})}{\partial u} = \frac{\partial f(\mathbf{u}, \mathbf{v})}{\partial v} = \mathbf{0} \quad (\text{Ec. 13})$$

Algoritmo de Horn y Schunck

Como fue explicado en secciones anteriores, Horn y Schunck utilizan dos consignas para la solución del problema de flujo óptico: la restricción de constancia de brillo y la restricción de suavidad. Esta segunda consigna añade un término global a la imagen. El algoritmo de Horn y Schunck resuelve la ecuación de diferencias por medio de la iteración de las derivadas parciales hasta converger hacia un error planteado.

Horn y Schunck plantean la minimización de los términos de sus consignas: el error ϵ_b , que hace referencia a la restricción de brillo, y el error ϵ_c^2 , que hace referencia a la restricción de suavidad, denotados por las ecuaciones 14 y 15 (Horn & Schunck, 1980).

$$\epsilon_b = E_x u + E_y v + E_t \quad (\text{Ec. 14})$$

$$\epsilon_c^2 = \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 \quad (\text{Ec. 15})$$

El error total que se busca minimizar es la suma de estas componentes, con un factor de peso α^2 , el cual cumple un rol importante netamente en regiones con un gradiente de brillo pequeño. El error que este método minimiza es el planteado en la ecuación 16 (Horn & Schunck, 1980).

$$\epsilon^2 = \iint (\epsilon_b + \alpha^2 \epsilon_c^2) dx dy \quad (\text{Ec. 16})$$

La ecuación 16 puede ser resuelta utilizando métodos de cálculo variacional, y los términos resultantes comprenden un Laplaciano (adicionalmente a las derivadas parciales espaciales y temporales). Se utilizan los métodos de aproximaciones analizados en la sección *Estimación de Derivadas por Diferencias Finitas y Detección de Bordos* para obtener las soluciones para el flujo óptico, como se muestra en las ecuaciones 17 y 18 (Horn & Schunck, 1980).

$$(E_x^2 + E_y^2)(u - \bar{u}) = -E_x[E_x \bar{u} + E_y \bar{v} + E_t] \quad (\text{Ec. 17})$$

$$(E_x^2 + E_y^2)(v - \bar{v}) = -E_y[E_x \bar{u} + E_y \bar{v} + E_t] \quad (\text{Ec. 18})$$

El algoritmo planteado por Horn y Schunck tiene una naturaleza iterativa, debido a que se busca minimizar los errores planteados hasta converger a un valor propuesto. Es por eso que las ecuaciones 17 y 18 se reestructuran para obtener soluciones iterativas (Horn & Schunck, 1980). Adicionalmente se puede abreviar los términos de las derivadas que se

mantienen constantes para ambas componentes del flujo óptico, de manera que $\mathbf{P} = \mathbf{E}_x \bar{\mathbf{u}} + \mathbf{E}_y \bar{\mathbf{v}} + \mathbf{E}_t$ y $\mathbf{D} = \mathbf{E}_x^2 + \mathbf{E}_y^2 + \alpha^2$ (se adiciona el factor de peso en este último término) (Shah M. , 2011). Las ecuaciones 19 y 20 muestran las soluciones iterativas resultantes

$$\mathbf{u}^{n+1} = \bar{\mathbf{u}}^n - \mathbf{E}_x \left(\frac{\mathbf{P}}{\mathbf{D}} \right) \quad (\text{Ec. 19})$$

$$\mathbf{v}^{n+1} = \bar{\mathbf{v}}^n - \mathbf{E}_y \left(\frac{\mathbf{P}}{\mathbf{D}} \right) \quad (\text{Ec. 20})$$

Algoritmo de Lucas y Kanade

A diferencia del método de Horn y Schunck, la segunda consigna utilizada por Lucas y Kanade para la determinación del flujo óptico es la restricción de gradiente. El algoritmo desarrollado por estos autores aplica una restricción de suavidad local (diseñada para manejar vecindarios pequeños) utilizando un ajuste de mínimos cuadrados en una pequeña ventana de la imagen para el análisis de la traslación del objeto en cuestión (Lucas & Take, 1981). El algoritmo de Lucas y Kanade resuelve la ecuación diferencial de constancia de brillo por medio de la optimización de un funcional energético. En la ecuación 21 se presenta el modelo matricial que resuelve este algoritmo, donde los subíndices en las derivadas representan el número del pixel analizado (Shah M. , 2011).

$$\begin{bmatrix} \mathbf{E}_{x_1} & \mathbf{E}_{y_1} \\ \mathbf{E}_{x_2} & \mathbf{E}_{y_2} \\ \vdots & \vdots \\ \mathbf{E}_{x_n} & \mathbf{E}_{y_n} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = - \begin{bmatrix} \mathbf{E}_{t_1} \\ \mathbf{E}_{t_1} \\ \vdots \\ \mathbf{E}_{t_1} \end{bmatrix} \quad (\text{Ec. 21})$$

Se representa en la ecuación 22 el modelo matricial de la ecuación 21 a manera de ecuación simplificada, donde \mathbf{A} representa la matriz de derivadas espaciales, \mathbf{u} el vector de velocidades, y \mathbf{E}_t las derivadas temporales (Shah M. , 2011).

$$\mathbf{u} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{E}_t \quad (\text{Ec. 22})$$

La ecuación 22 se resuelve realizando un ajuste de mínimos cuadrados, posterior a lo cual se utiliza cálculo variacional para encontrar una “representación estructural de Tensores” (Bagci, 2015), como se muestra en la ecuación 23.

$$\begin{bmatrix} T_{xx} & T_{xy} \\ T_{xy} & T_{yy} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} = - \begin{bmatrix} T_{xt} \\ T_{yt} \end{bmatrix} \quad (\text{Ec. 23})$$

En la ecuación 23 se definen a los tensores como $T_{xx} = \sum E_x^2$, $T_{yy} = \sum E_y^2$, $T_{xy} = \sum E_x E_y$, $T_{xt} = \sum E_x E_t$ y $T_{yt} = \sum E_y E_t$. De esta definición se pueden encontrar las componentes del flujo óptico por medio de las ecuaciones 24 y 25 (Bagci, 2015).

$$\mathbf{u} = \frac{T_{yt}T_{xy} - T_{xt}T_{yy}}{T_{xx}T_{yy} - T_{xy}^2} \quad (\text{Ec. 24})$$

$$\mathbf{v} = \frac{T_{xt}T_{xy} - T_{yt}T_{xx}}{T_{xx}T_{yy} - T_{xy}^2} \quad (\text{Ec. 25})$$

En el algoritmo de Lucas y Kanade se utiliza generalmente segmentos pequeños de una imagen (ventana o vecindario), sin embargo su aplicación no se limita a ello (Collins, 2007). Es por ello que para la implementación de este algoritmo se utilizó el siguiente algoritmo:

1. Se dividió la imagen en vecindarios pequeños de d dimensiones.
2. Se calculó el flujo óptico para para cada vecindario.
3. Se generalizó el flujo óptico para cada pixel correspondiente al vecindario tratado.
4. Se recuperó el tamaño original utilizando el flujo óptico generalizado.

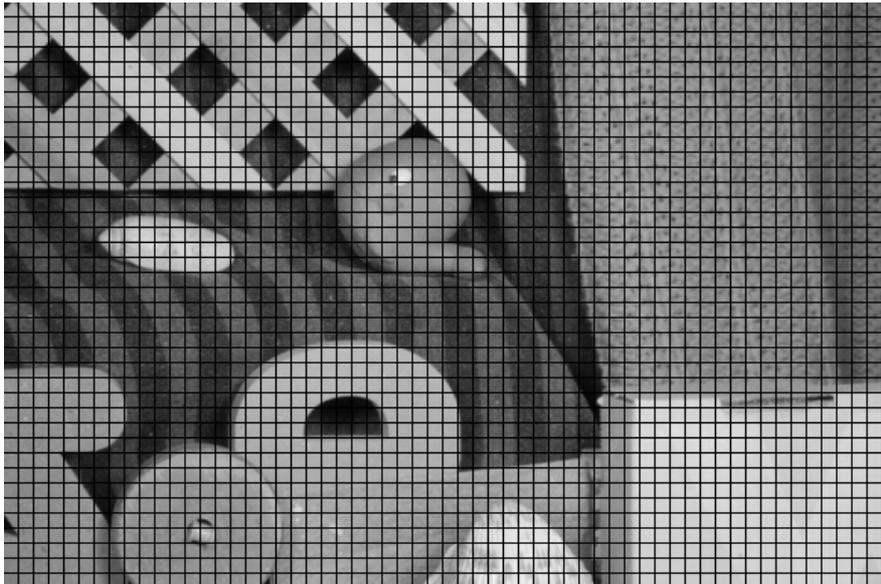
Resultados

Conforme a las consignas planteadas por cada uno de los autores se implementaron ambos algoritmos para comparar su funcionamiento. En los algoritmos implementados se variaron los parámetros característicos hasta obtener los mejores resultados. En las siguientes

secciones subsiguientes se presenta un análisis de cada método y el proceso seguido hasta la obtención de resultados óptimos.

Para realizar las pruebas necesarias se utilizó la secuencia de imágenes mostrada en la Figura 7.

Frame 1



Frame 2

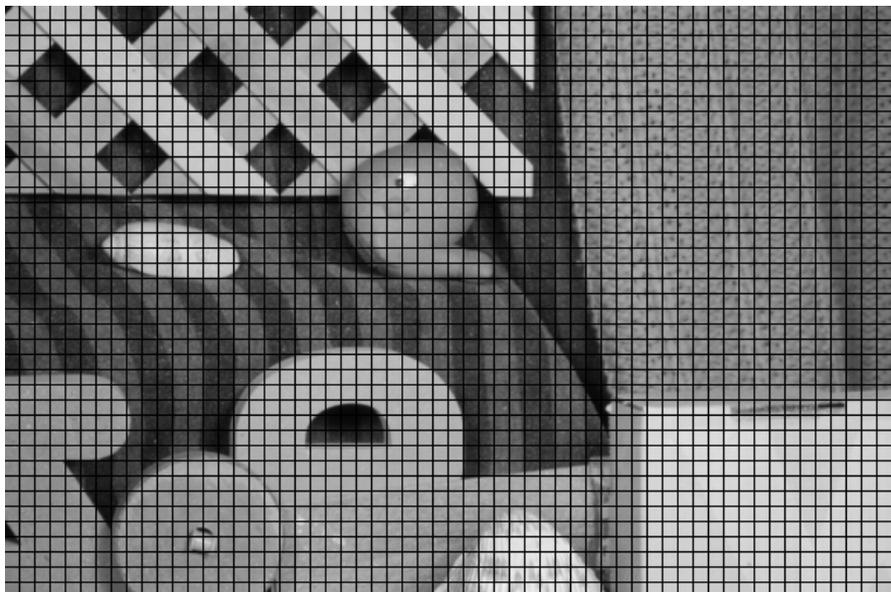


Figura 7: Dos imágenes consecutivas de la secuencia de video “RubberWhale”, editada en Matlab para tener una malla de referencia.

(Tomado de: Middlebury. (2009). *Optical Flow*. Middlebury, Estados Unidos. Obtenido el 31 de Enero del 2016 de <http://vision.middlebury.edu/flow/data/>)

Se muestran en las imágenes resultantes para ambos métodos el flujo óptico plasmado sobre el primer cuadro de la secuencia analizada para poder observar claramente la distribución de movimiento en la secuencia original. También se muestra el flujo óptico sin la secuencia por detrás.

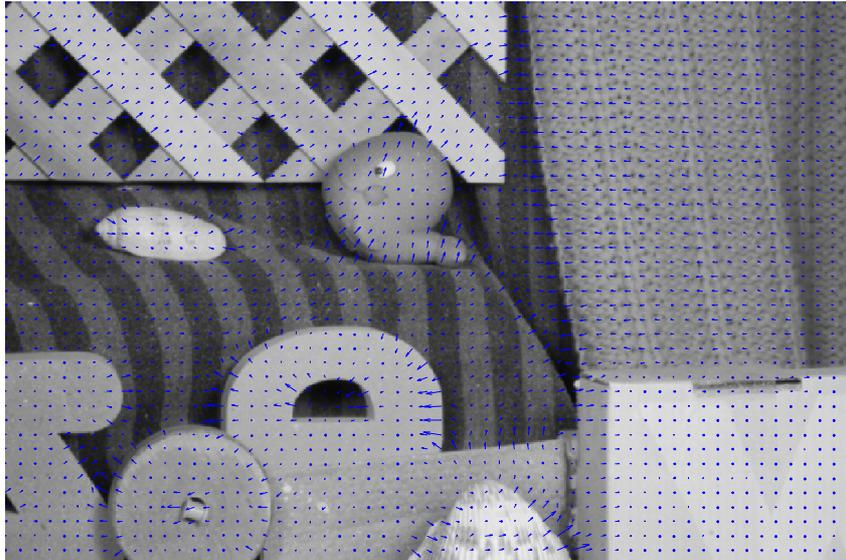
Resultados del algoritmo de Horn & Schunck

En la Tabla 1 se presenta los parámetros que varían y la figura correspondiente a los resultados para el algoritmo de Horn y Schunck. Se representa como " N° " el número de la prueba realizada, N_{it} el número de iteraciones que utiliza el algoritmo, α el peso descrito en secciones anteriores, t_{prom} el tiempo promedio del lazo más importante del algoritmo, t_{tot} el tiempo total que utiliza el algoritmo en correr y "**Resultado**" muestra el número de la figura correspondiente a los valores planteados.

Tabla 1: Parámetros variados y resultados de algoritmo implementado de Horn y Schunck.

N°	N_{it}	α	t_{prom} [s]	t_{tot} [s]	Resultado
1	100	100	3.37	343.51	Fig 8
2	100	50	3.36	341.25	Fig 9
3	25	200	3.35	89.44	Fig 10
4	25	2	3.34	90.57	Fig 11
5	2	200	3.37	11.66	Fig 12
6	2	2	3.36	11.71	Fig 13

Frame 1 Flujo Óptico



Flujo Óptico

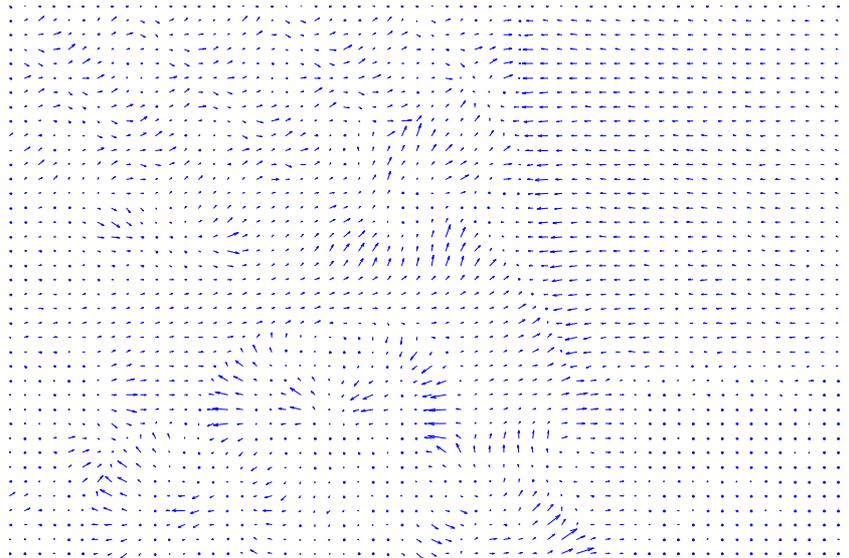
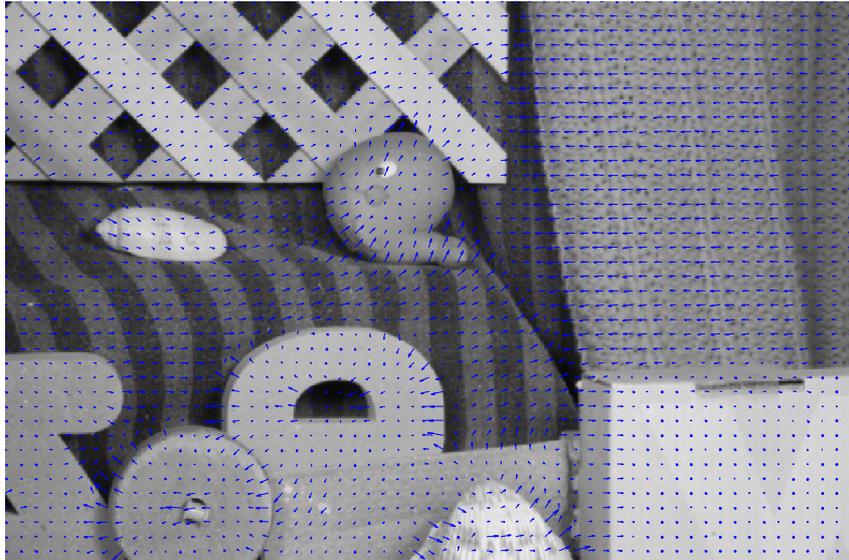
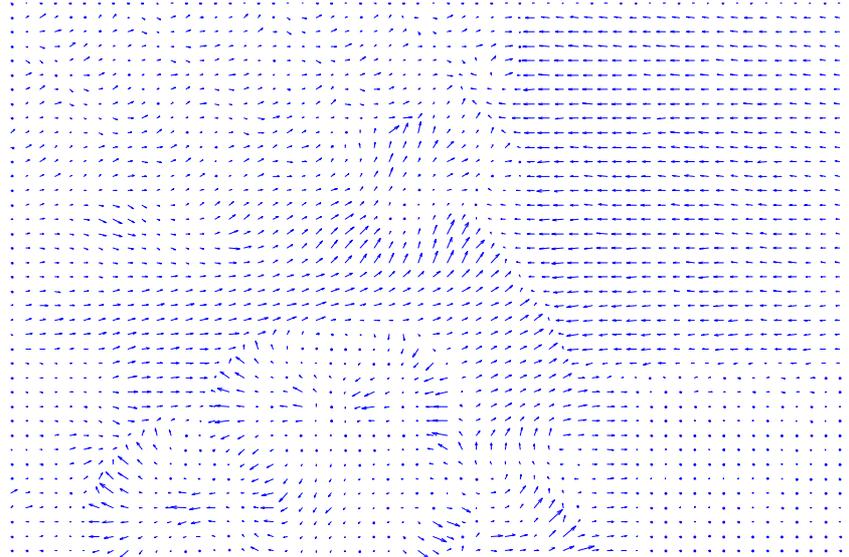


Figura 8: Resultado para Horn y Schunck con $N_{it} = 100$ y $\alpha = 100$.

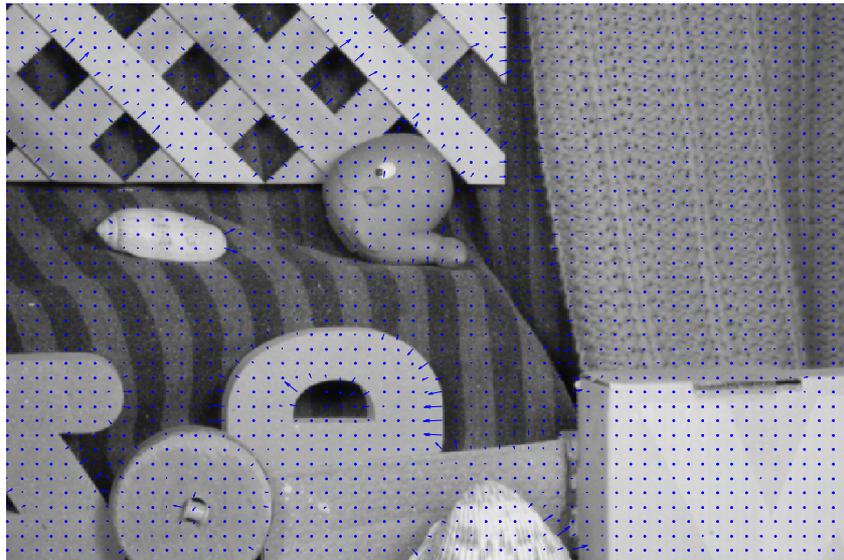
Frame 1 Flujo Óptico



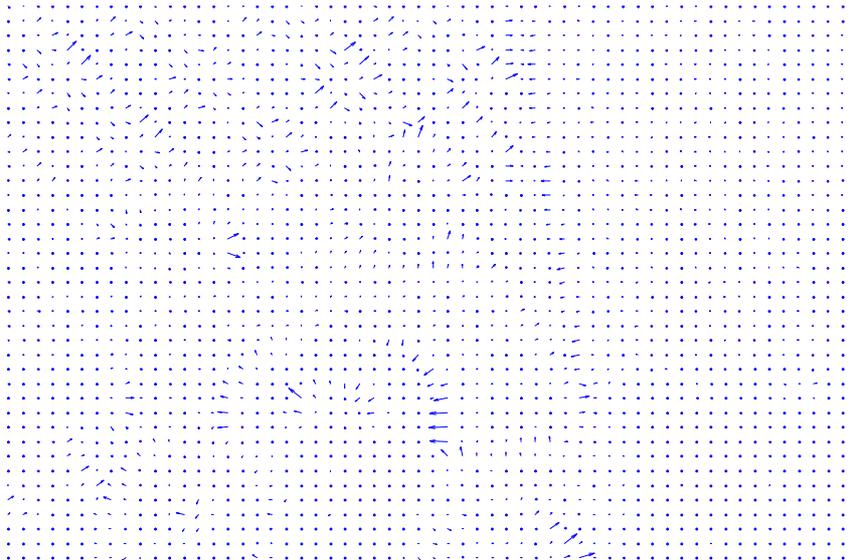
Flujo Óptico

**Figura 9:** Resultado para Horn y Schunck con $N_{it} = 100$ y $\alpha = 50$.

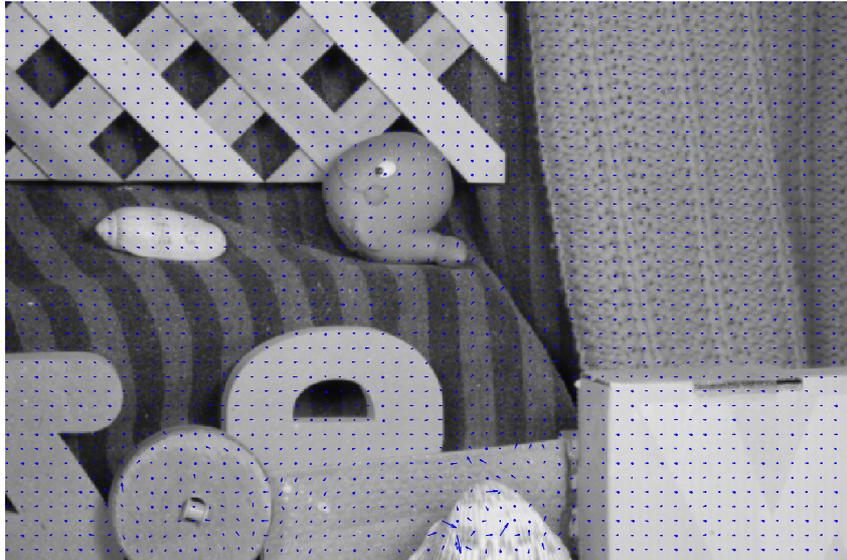
Frame 1 Flujo Óptico



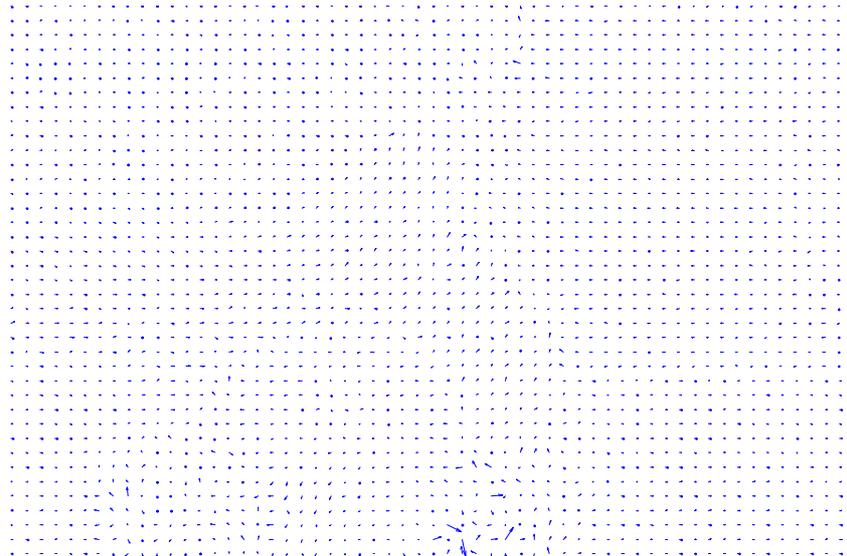
Flujo Óptico

**Figura 10:** Resultado para Horn y Schunck con $N_{it} = 25$ y $\alpha = 200$.

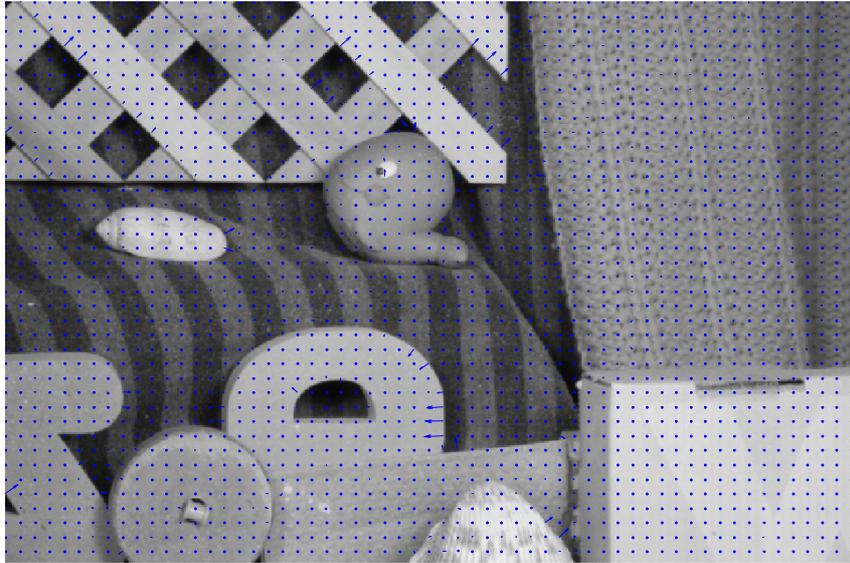
Frame 1 Flujo Óptico



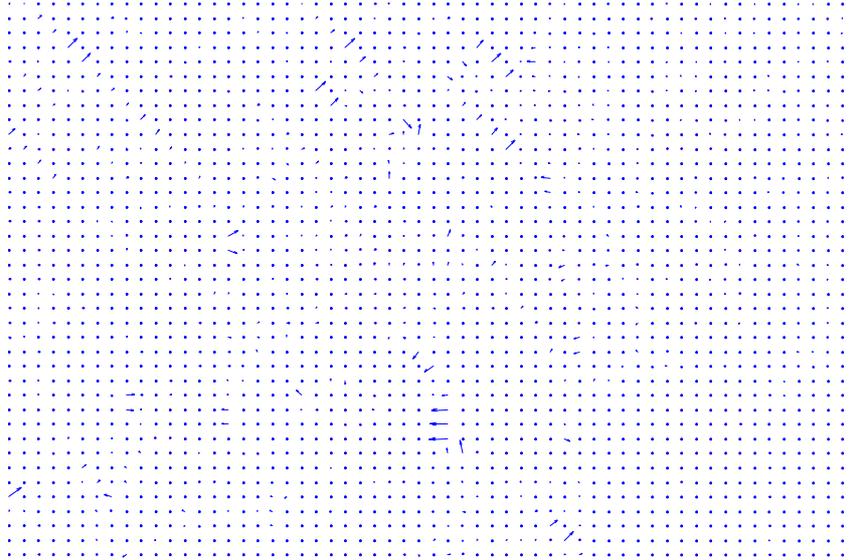
Flujo Óptico

**Figura 11:** Resultado para Horn y Schunck con $N_{it} = 25$ y $\alpha = 2$.

Frame 1 Flujo Óptico



Flujo Óptico

**Figura 12:** Resultado para Horn y Schunck con $N_{it} = 2$ y $\alpha = 200$.

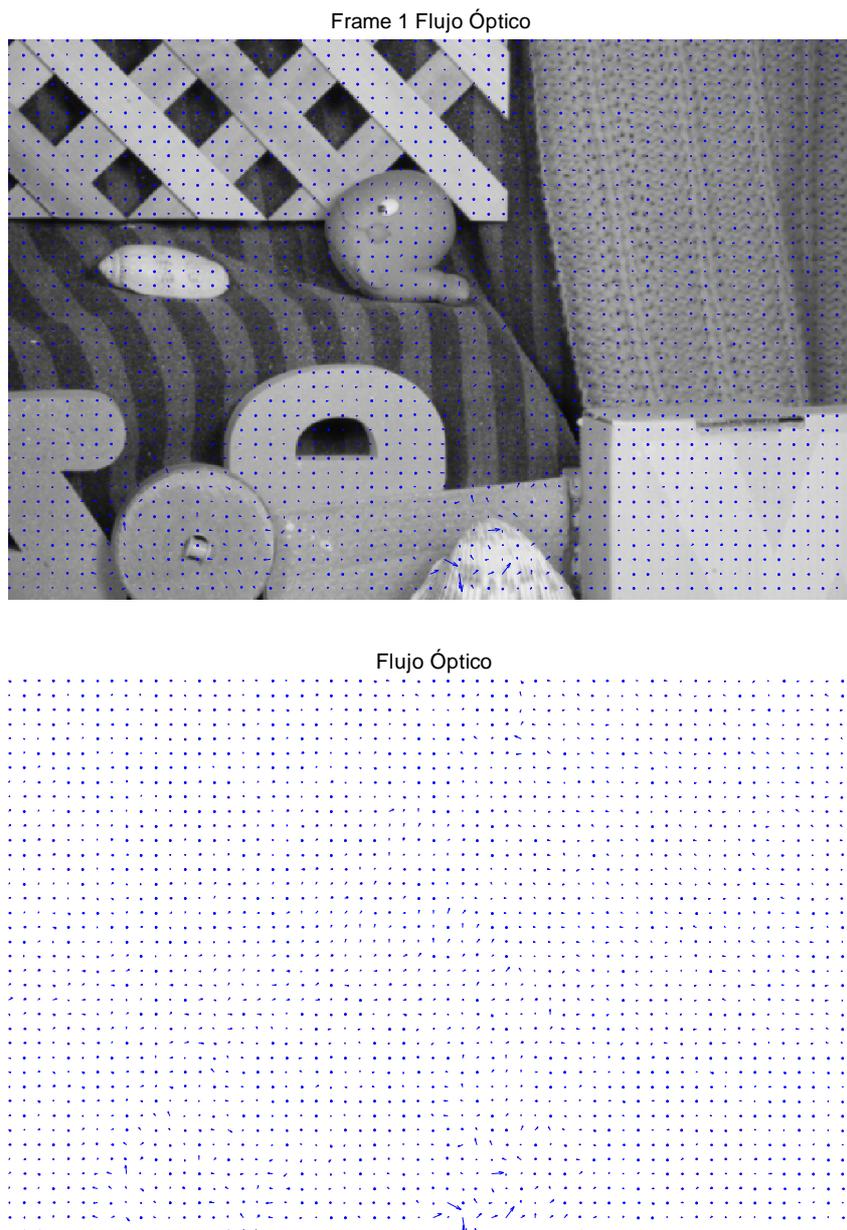


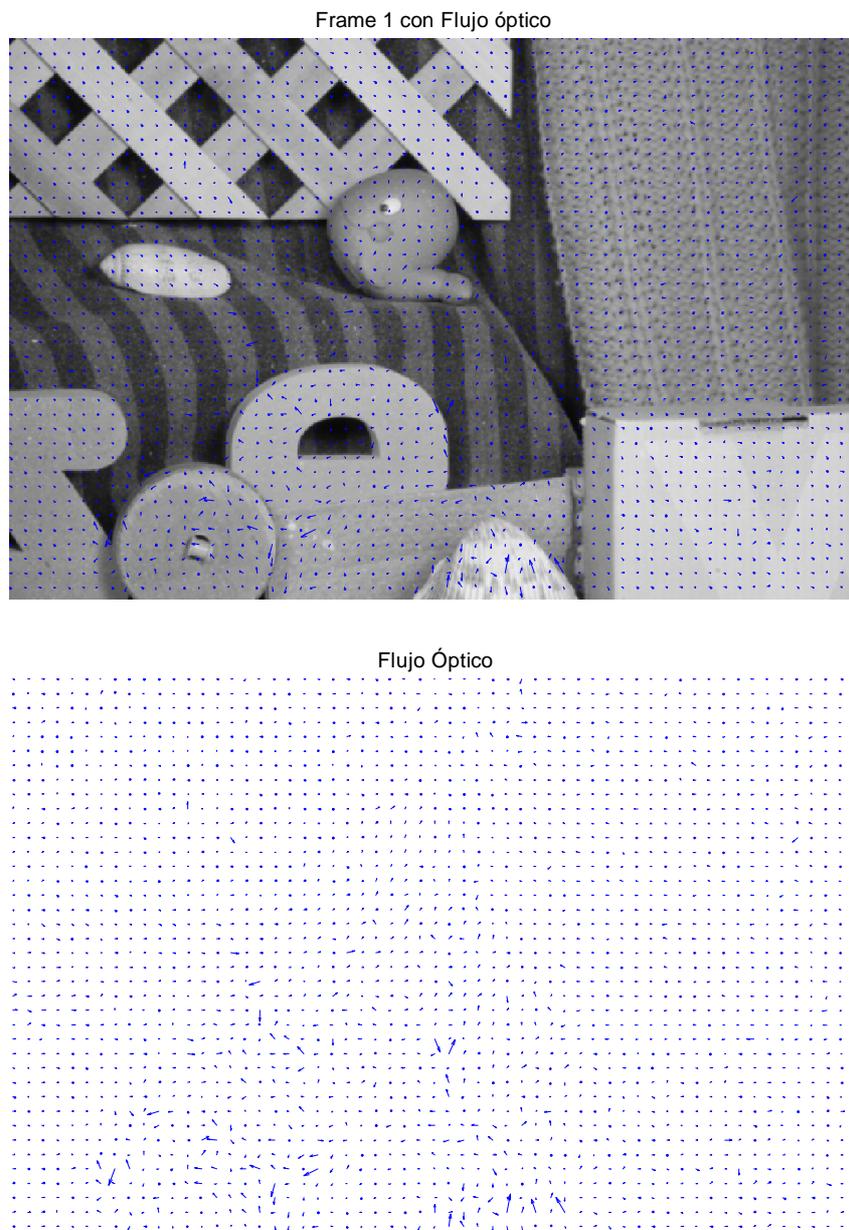
Figura 13: Resultado para Horn y Schunck con $N_{it} = 2$ y $\alpha = 2$.

Resultados del algoritmo de Lucas y Kanade

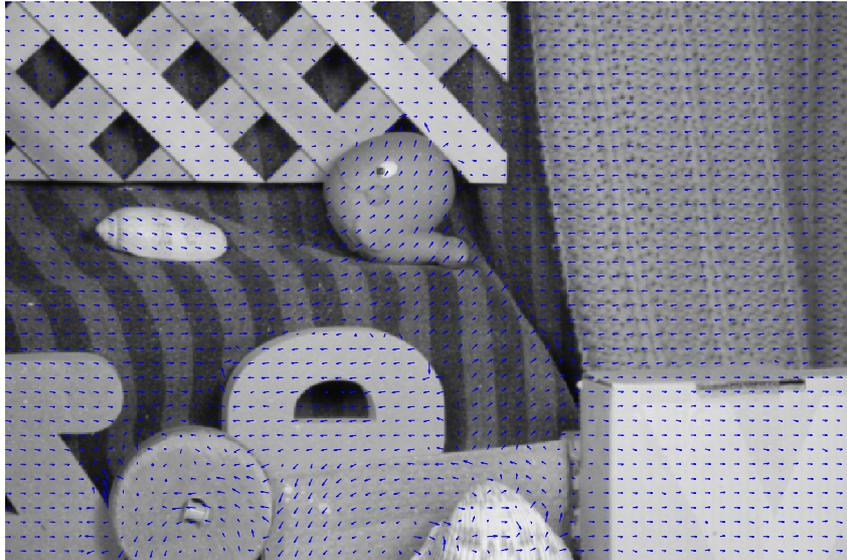
Se presenta en la Tabla 2 los parámetros variados y la figura correspondiente a los resultados para el algoritmo de Lucas y Kanade. Se representa como " N^o " el número de la prueba realizada, "**Pasos**" es la dimensión del vecindario, t_{prom} el tiempo promedio del lazo más importante del algoritmo, t_{tot} el tiempo total que utiliza el algoritmo en correr y "**Resultado**" muestra el número de la figura correspondiente a los valores planteados

Tabla 2: Parámetros variados y resultados de algoritmo implementado de Lucas y Kanade.

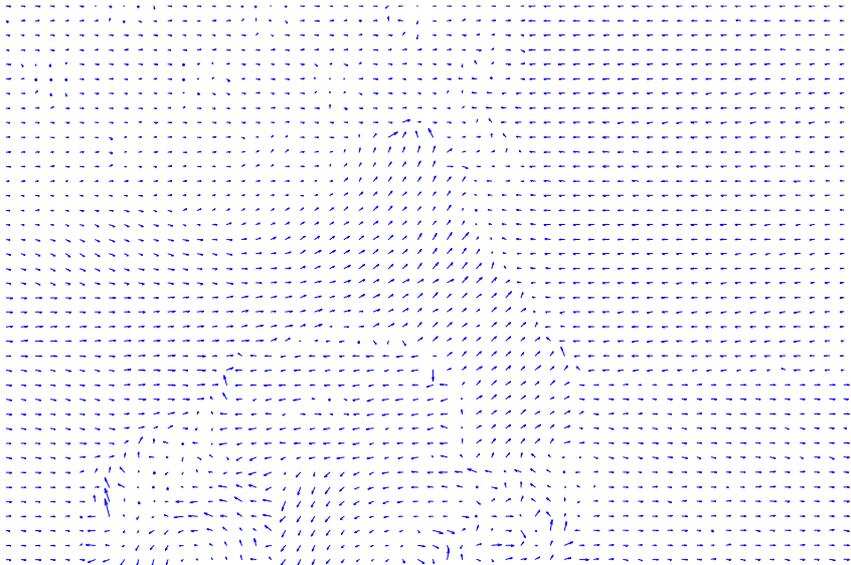
N°	<i>Pasos</i> [píxeles]	t_{prom} [s]	t_{tot} [s]	<i>Resultado</i>
1	2	0.01	3.82	Fig 14
2	8	0.02	2.21	Fig 15
3	32	0.03	2.14	Fig 16
4	64	0.09	2.11	Fig 17
5	80	0.12	2.09	Fig 18
6	100	0.14	2.08	Fig 19

**Figura 14:** Resultado para Lucas y Kanade con *Pasos* = 2.

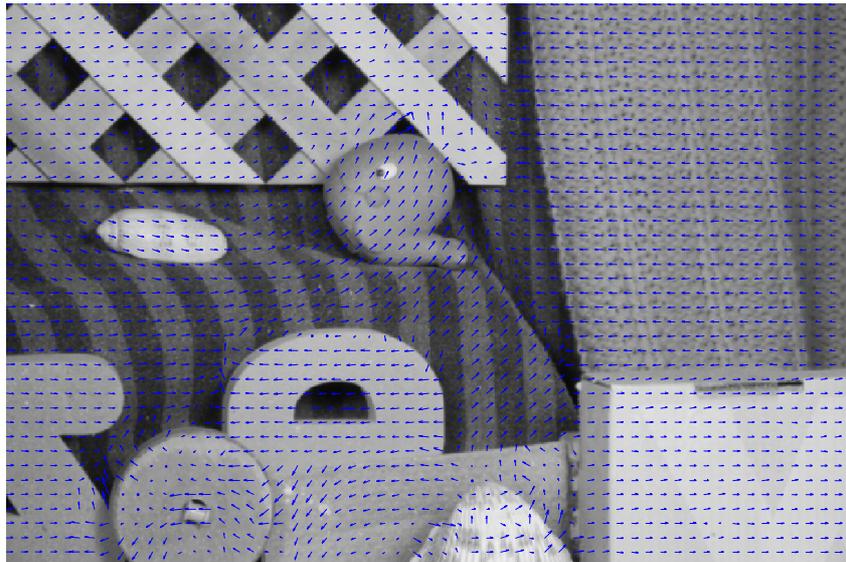
Frame 1 con Flujo óptico



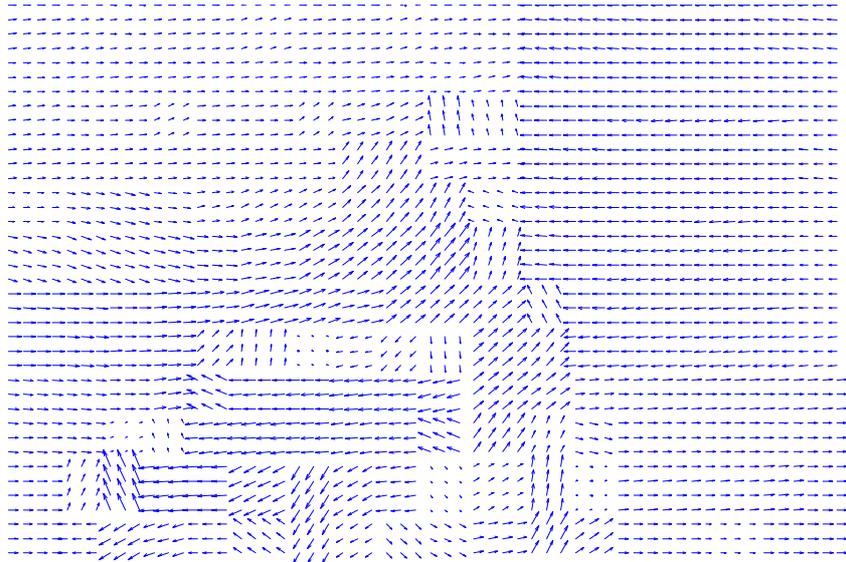
Flujo Óptico

**Figura 15:** Resultado para Lucas y Kanade con *Pasos* = 8.

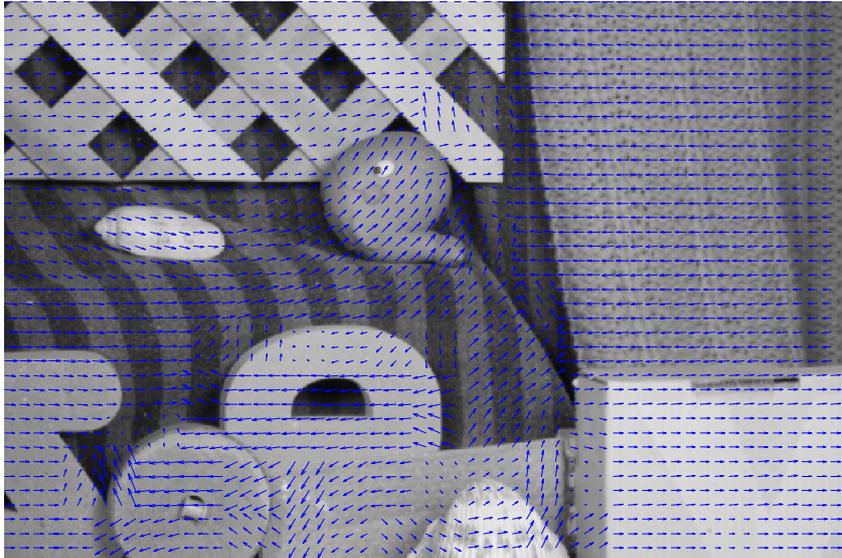
Frame 1 con Flujo óptico



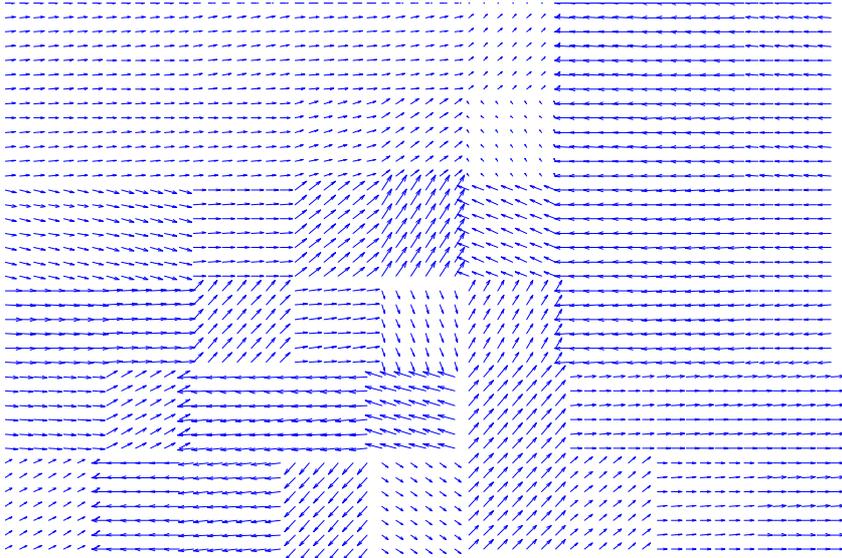
Flujo Óptico

**Figura 16:** Resultado para Lucas y Kanade con *Pasos* = 32.

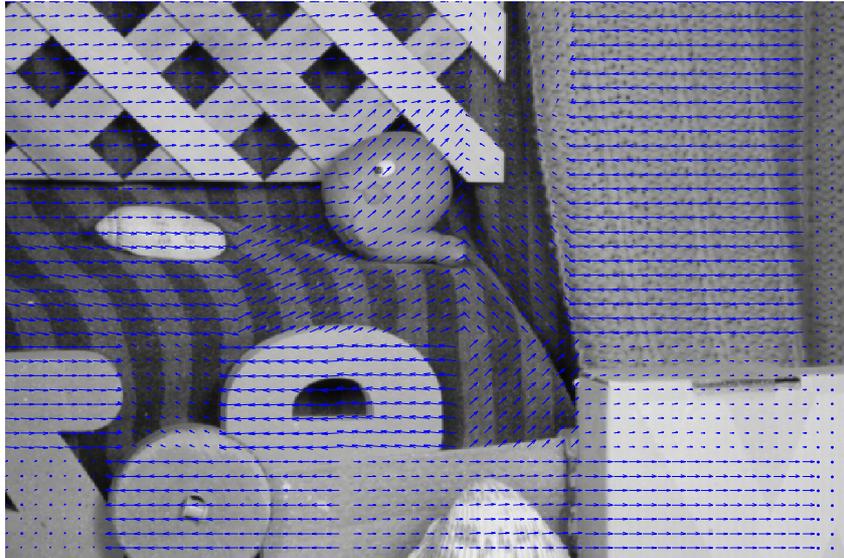
Frame 1 con Flujo óptico



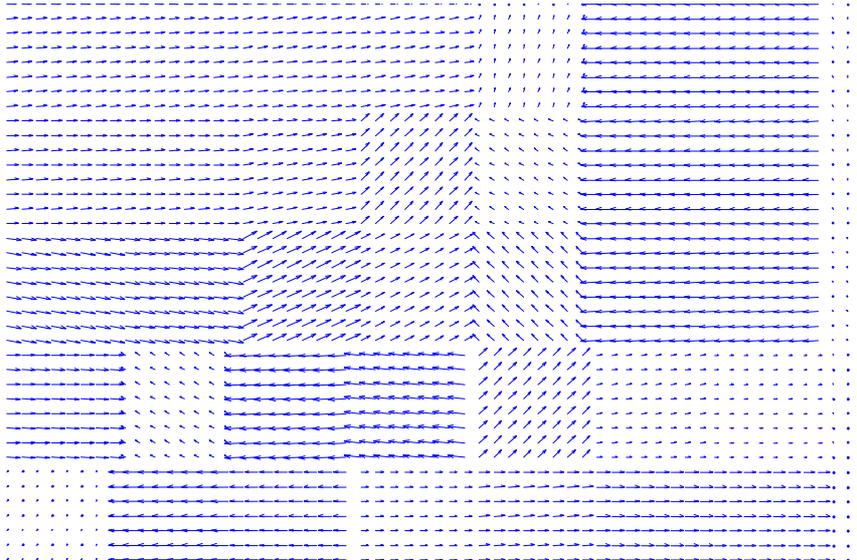
Flujo Óptico

**Figura 17:** Resultado para Lucas y Kanade con *Pasos* = 64.

Frame 1 con Flujo óptico



Flujo Óptico

**Figura 18:** Resultado para Lucas y Kanade con *Pasos* = 80.

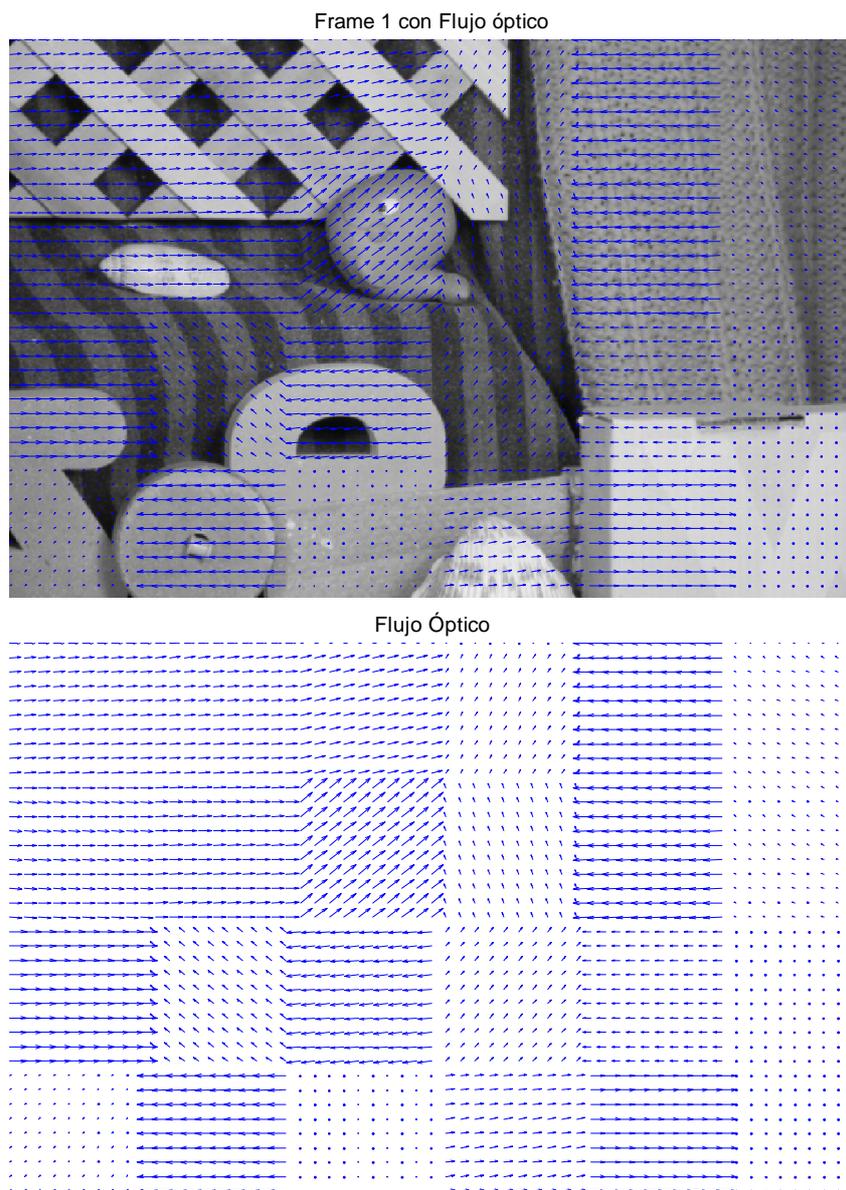


Figura 19: Resultado para Lucas y Kanade con ***Pasos* = 100**.

Análisis de Resultados

Los resultados obtenidos para el método de Horn y Schunck muestran que el parámetro N_{it} (número de iteraciones) define la calidad del cálculo de flujo óptico para este método, siendo un mayor número de iteraciones la mejor opción. Adicionalmente, en la subsección *Algoritmo de Horn y Schunck* se expone el uso de la variable α dentro de las consideraciones de cálculo de este método. El mejor valor para asignar a α depende de las

características de la imagen (existencia de textura en los objetos y comparación de ésta con la variación de brillo). Al observar las ecuaciones 19 y 20 se puede concluir que un factor α muy bajo conlleva a una mayor importancia en la consigna de suavidad, y ocurre lo contrario con un factor α bajo. Sin embargo, al realizar varias pruebas para el algoritmo de Horn y Schunck se observa la importancia de la presencia de texturas y, por tanto, la importancia de encontrar un valor para α adecuado conforme a las características específicas de la imagen.

Conforme a los resultados obtenidos para el algoritmo de Lucas y Kanade, el parámetro *Pasos* afecta directamente a la estructura de los vectores del flujo óptico al momento de reconstruir el campo vectorial total de la imagen, el cual tiene las mismas dimensiones que la misma. Acorde a las pruebas realizadas, el mejor resultado fue obtenido con un tamaño de vecindario (parámetro *Pasos*) relativamente pequeño, como en la prueba de *Pasos* = 8 píxeles (Figura 15). Sin embargo, al ajustar este parámetro en dos píxeles el algoritmo no capta movimiento (en la mayoría de los casos). Esto se debe a que cuando se consideran vecindarios de dimensiones muy pequeñas la mayoría de ellos no presentan un cambio de brillo, como se muestra en la Figura 20 **a.**, dando como resultado un flujo óptico igual a cero. Al incrementar las dimensiones de los vecindarios el algoritmo comienza a captar cambios en el brillo (Figura 20 **b.**) lo que se plasma en el flujo óptico del vecindario analizado. Esto permite captar el movimiento de patrones de brillo medianamente grandes de cuadro a cuadro. No obstante, vecindarios con dimensiones muy grandes (Figura 20 **b.**), como en la prueba realizada con *Pasos* = 100 píxeles (Figura 20 **c.**), abarcan demasiados cambios de brillo en la ventana, de manera que el flujo óptico se generaliza para todo el vecindario y la estimación de movimiento de los objetos se pierde.

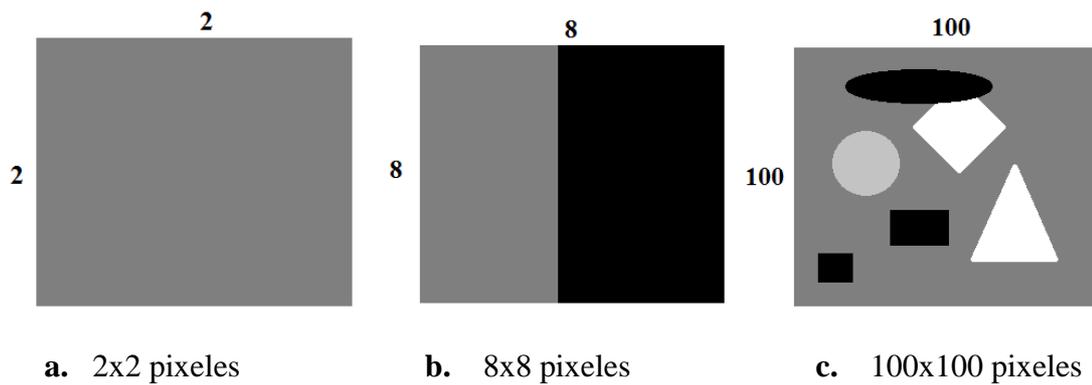


Figura 20: Imágenes de muestra para comprensión de diferencias conforme a cambios en dimensiones de vecindarios. Se muestran vecindarios con diferentes medidas tomados de una imagen.

CONCLUSIONES

El análisis presentado en este trabajo muestra las características de resolución del problema de constancia de brillo para el cálculo de flujo óptico por medio de dos métodos diferentes. Al observar los resultados obtenidos y comparar el parámetro de tiempo de ejecución de cada algoritmo (conforme se muestra en las Tablas 1 y 2) se puede observar la rapidez con la que trabaja el algoritmo planteado por Lucas y Kanade, frente al de Horn y Schunck, lo que deriva de la naturaleza iterativa del método desarrollado por estos últimos investigadores. Desde el punto de vista algorítmico, el tiempo que tardan los métodos en converger a un error mínimo para obtener los resultados más óptimos está directamente relacionado a la cantidad de lazos que comprenda su estructura y al tiempo que tarde el programa en resolver las ecuaciones comprendidas en estos lazos. Se muestra en la Tabla 1 los tiempos promedio de resolución del lazo de mayor trascendencia en el algoritmo de Horn y Schunck. Al comparar los tiempos que tardan en resolver el problema del flujo óptico ambos algoritmos, se observa una diferencia muy drástica que se refleja en la ventaja del modelo tensorial frente al iterativo.

Al referirse a la estructura, el algoritmo desarrollado por Lucas y Kanade tiene la ventaja de la resolución directa de una ecuación matricial derivada de la ecuación de constancia de brillo, lo que implica un costo computacional significativamente menor comparándolo con la resolución del algoritmo de Horn y Schunck. Sin embargo, el algoritmo de Lucas y Kanade disminuye en cierta cantidad su grado de eficiencia al ser un método que considera una constancia de velocidades en una ventana pequeña, requiriendo una división de la imagen en vecindarios de pequeñas dimensiones para posteriormente realizar el cálculo de flujo óptico por vecindario y finalmente generalizar el flujo a toda esta ventana reconstruyendo de esta manera el campo vectorial total. A pesar de este no tan insignificante

trabajo de división y reagrupación de vecindarios, el algoritmo de Lucas y Kanade se muestra más eficiente en cuestión de estructura que el de Horn y Schunck.

Los resultados obtenidos para el algoritmo de Horn y Schunck muestran cómo el flujo óptico varía conforme el parámetro α cambia. Esto depende de la naturaleza de la imagen, como se analiza en la sección de *Resultados*. Para las pruebas realizadas sobre la secuencia de “RubberWhale” el valor de α que comprendió los mejores resultados fue el de $\alpha = 50$, y el valor de N_{it} que describió un mejor flujo óptico fue el de $N_{it} = 100$ (número máximo de iteraciones planteado para las pruebas), como se muestra en la Figura 9.

Conforme a los resultados obtenidos en las pruebas realizadas con el algoritmo de Lucas y Kanade, la dimensión de los vecindarios para los cuales los resultados fueron óptimos fue $Pasos = 8$, debido a que se capta el movimiento de los objetos más significantes en la imagen y no se generaliza el flujo óptico para objetos o fondos adyacentes a estos objetos. Sin embargo, con este parámetro planteado el movimiento de las texturas fue prácticamente despreciado.

Trabajos posteriores se han realizado en el campo del rastreo de objetos y visión por computadora. La implementación de métodos de refinación *grueso a fino* para los algoritmos de flujo óptico ahora forma una parte esencial en éstos. Sin embargo, estos métodos para mejorar la eficiencia de los algoritmos han sido excluidos del presente trabajo de comparación para poder observar las características planteadas por las restricciones que han sido consideradas por cada autor en su origen. En posteriores trabajos, la implementación de métodos de pirámide de imágenes puede ser de gran utilidad y mejorar la eficiencia del algoritmo de Horn y Schunck, reduciendo de esta manera la cantidad de iteraciones necesarias para obtener resultados óptimos.

REFERENCIAS BIBLIOGRÁFICAS

- Bagci, U. (2015). *CAP 5415 - Computer Vision, Lecture 8- Optical Flow*. Orlando, Florida, Estados Unidos. Obtenido el 13 de Febrero del 2016 de <http://www.cs.ucf.edu/~bagci/teaching/computervision15/lec8.pdf>
- Barron, J., Fleet, D., & Beauchemin, S. (1994). *Systems and experiment : Performance of optical*. Ontario: Int. Journal of Computer Vision. Obtenido el 16 de Abril del 2016 de Obtenido el 22 de Abril del 2016 de <http://www.cs.toronto.edu/~fleet/research/Papers/ijcv-94.pdf>
- Collins, R. (2007). *Lecture 30-Video Tracking: Lucas-Kanade*. Pensilvania, Estados Unidos. Obtenido el 13 de Febrero del 2016 de <http://www.cse.psu.edu/~rtc12/CSE486/lecture30.pdf>
- Forsyth, D., & Ponce, J. (2012). *Computer Vision, A Modern Approach*. Pearson Education. Obtenido el 22 de Abril del 2016 de https://cdn.preterhuman.net/texts/science_and_technology/artificial_intelligence/Computer%20Vision%20A%20Modern%20Approach%20-%20Forsyth%20,%20Ponce.pdf
- Horn, B., & Schunck, B. (1980). *Determining Optical Flow*. Cambridge: North-Holland. Obtenido el 31 de Enero del 2016 de http://image.diku.dk/imagecanon/material/HornSchunckOptical_Flow.pdf
- Lucas, B., & Take, K. (1981). *An Iterative Image Registration Technique with an Application to Stereo Vision*. Vancouver. Obtenido el 31 de Enero del 2016 de https://www.ri.cmu.edu/pub_files/pub3/lucas_bruce_d_1981_1/lucas_bruce_d_1981_1.pdf

- Mallot, H., & Allen, J. (2000). *Computational Vision, Information Processing in Perception and Visual Behavior*. MIT Press. Obtenido el 28 de Mayo del 2016 de <http://www.federaljack.com/ebooks/Consciousness%20Books%20Collection/Hanspet%20er%20A.%20Mallot%20John%20S.%20Allen%20-%20Computational%20Vision.pdf>
- Middlebury. (2009). *Optical Flow*. Middlebury, Estados Unidos. Obtenido el 31 de Enero del 2016 de <http://vision.middlebury.edu/flow/data/>
- Shah, M. (1997). *Fundamentals of Computer Vision*. Orlando. Obtenido el 13 de Febrero del 2016 de <http://www.cse.unr.edu/~bebis/CS485/Handouts/ShahBook.pdf>
- Shah, M. (2011). *CAP 5415 - Computer Vision*. Orlando, Florida, Estados Unidos. Obtenido el 13 de Febrero del 2016 de <http://crcv.ucf.edu/courses/CAP5415/Fall2012/Lecture-6b-OpticalFlow.pdf>
- Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Washington. Obtenido el 22 de Marzo del 2016 de http://szeliski.org/Book/drafts/SzeliskiBook_20100903_draft.pdf