

Adición de un Paso de Fusión de Datos al Algoritmo de Mínimos Cuadrados Clásico para Mejorar la Estima del Campo de Flujo Óptico

Carlos M.Soria^a, Mário Sarcinelli-Filho^{b,*}, Ricardo Carelli^a

^a*Instituto de Automática, Facultad de Ingeniería, Universidad Nacional de San Juan, Av. Libertador San Martín Oeste 1109, 05400 San Juan - Argentina.*

^b*Departamento de Ingeniería Eléctrica, Universidad Federal del Espíritu Santo, Av. Fernando Ferrari 514, Vitoria, ES - Brasil.*

Resumen

En este artículo se propone una versión mejorada del método clásico basado en el criterio de mínimos cuadrados para estimar el campo de flujo óptico. La esencia de la propuesta es fusionar algunas estimas de un mismo vector de flujo óptico, obtenidas por el criterio de mínimos cuadrados, para generar una estima más precisa de dicho vector. Se discuten dos propuestas para obtener las estimas a fusionar en base al criterio clásico de mínimos cuadrados, generando dos versiones modificadas del algoritmo clásico. Se presentan los resultados obtenidos aplicándose los dos algoritmos propuestos para estimar los vectores de flujo óptico correspondientes a dos imágenes sucesivas de la secuencia de vídeo conocida como *SRI Trees*. Finalmente se enumeran algunas situaciones, en términos de navegación de robots móviles, en las cuales dichos algoritmos podrían ser utilizados, como forma de resaltar la aplicabilidad de los resultados de esta investigación. *Copyright © 2013 CEA. Publicado por Elsevier España, S.L. Todos los derechos reservados.*

Palabras Clave:

Flujo óptico, Fusión de datos, Estimación óptima, Estimación por mínimos cuadrados, Robots móviles autónomos.

1. Introducción

El problema de procesar secuencias de imágenes para calcular el flujo óptico ha sido estudiado por muchos años. De forma paralela, se han publicado varios informes acerca del uso de flujo óptico en diversas aplicaciones, como, por ejemplo, la recuperación de información de la escena tridimensional y parámetros de movimiento con una cámara en movimiento (Zhang and Faugeras, 1992; Zheng and Chellappa, 1993), la segmentación de movimiento (Black and Anandan, 1990), el cálculo de la disparidad estereó (Langley et al., 1991), la medición del flujo sanguíneo y de la movilidad de las paredes del corazón en las imágenes médicas (Prince and McVeigh, 1992), y la auto-conducción de vehículos (Giachetti et al., 1998; de Michelini and Verri, 1993; Gern et al., 2002; Zainal et al., 2012). El flujo óptico también ha sido utilizado en la navegación de robots móviles (Sarcinelli-Filho et al., 2002; Dev et al., 1997; Carelli et al., 2002; Jia et al., 2008; Caldeira et al., 2007), incluyendo vehículos aéreos autónomos (Rudall, 2004; Watanabe et al., 2009). Además, hay informes recientes de aplicaciones

que utilizan flujo óptico, como es el caso de los trabajos de Jia et al. (2008), Watanabe et al. (2009), Caldeira et al. (2007), Simas et al. (2010) y Zainal et al. (2012). Por lo tanto, un método eficiente para determinar el flujo óptico es de interés en muchas aplicaciones. En este contexto, este trabajo propone un algoritmo para calcular el campo de flujo óptico, que es una modificación del método clásico de Lukas and Kanade (1981), en el cual cada vector de flujo óptico se estima mediante un criterio de mínimos cuadrados.

El flujo óptico es una representación conveniente y útil para el campo de movimiento 3-D en el plano de la imagen (2D). Se pueden utilizar distintas técnicas para lograr obtener el flujo óptico. La mayoría de estas técnicas, sin embargo, requieren un gran esfuerzo de cómputo para calcular el campo de flujo óptico, exigen una gran capacidad de almacenamiento de imágenes, o generan valores inexactos. Por lo tanto, la elección de un algoritmo adecuado para calcular el campo de flujo óptico es una cuestión de compromiso entre estos tres aspectos, siempre que se cumplan con los requisitos de la aplicación deseada. En navegación de robots móviles, la principal motivación de este trabajo, los ordenadores disponibles a bordo del robot son a menudo insuficientes para el cálculo intensivo, y tienen una capacidad limitada para almacenar imágenes. El primer paso, entonces, es limitar el número de cuadros de imagen utilizados al mínimo necesario, es decir, dos cuadros de imagen sucesivos.

* Autor en correspondencia

Correos electrónicos: csoria@inaut.unsj.edu.ar (Carlos M.Soria), mario.sarcinelli@ele.ufes.br (Mário Sarcinelli-Filho), rcarelli@inaut.unsj.edu.ar (Ricardo Carelli)

Así es que muchos de los métodos disponibles en la literatura no son buenos candidatos para calcular el flujo óptico a bordo de vehículos autónomos, especialmente cuando se trata de vehículos de pequeño porte. En este contexto, se debe mencionar un algoritmo clásico para estimar el campo de flujo óptico, el cual solamente utiliza dos cuadros de imagen y exige esfuerzo computacional reducido, generando un campo de flujo óptico denso (Barron and Beauchemin, 1994). Dicho método, conocido como método de Lukas and Kanade (1981), por los nombres de sus proponentes, genera una estimación basada en mínimos cuadrados para un vector que representa el flujo óptico en una región limitada del cuadro de imagen bajo análisis, en la cual se supone que el flujo óptico es constante. Dicha estimación del vector de flujo óptico es asociada a una varianza, para expresar cuán precisa es la misma. Por lo tanto, este método es compatible con el filtrado de estadísticas, que permite reducir la incertidumbre asociada al vector de flujo óptico obtenido. Conviene señalar que este método aún se utiliza ampliamente (Carelli et al., 2002; Simas et al., 2010), debido a su bajo costo de cómputo y por generar un campo de flujo óptico denso (Barron and Beauchemin, 1994).

En este contexto, lo que se propone en este trabajo es obtener un conjunto de estimaciones iniciales de cada vector de flujo óptico y fusionarlas, para producir una estimación óptima de cada vector de flujo óptico (óptima en el sentido de que su varianza será menor que la menor de las varianzas asociadas a las estimaciones iniciales (Hong, 1999)). Así, la contribución del artículo es agregar un paso de fusión de datos al algoritmo clásico de Lukas and Kanade (1981) para obtener estimaciones más precisas de los vectores de flujo óptico que componen el campo de flujo óptico correspondiente a una imagen dada.

La primera parte del artículo describe el algoritmo para calcular el campo de flujo óptico con mayor precisión, compuesto por un paso correspondiente al cálculo del algoritmo original de Lukas and Kanade (1981) y por un paso de fusión de datos. Las estimaciones iniciales de los vectores de flujo óptico que se fusionarán se calculan utilizando las derivadas espacio-tiempo en todos los pixeles de la región considerada, de la misma forma que el algoritmo original de Lukas and Kanade (1981).

A continuación, se cambia la forma como se calculan las estimaciones iniciales de cada vector de flujo óptico, admitiéndose una pérdida de precisión para reducir el tiempo de cálculo. Esto se realiza reduciendo el número de pixeles utilizados en la obtención de las estimaciones iniciales de los vectores de flujo óptico, similar a lo que se hace en Caldeira et al. (2007), por ejemplo. Esto resulta en una segunda versión modificada del algoritmo de Lucas y Kanade, suficientemente rápida para aplicaciones en navegación de robots móviles, especialmente en situaciones en que no se requiere mucha precisión (por ejemplo, evasión de obstáculos (Sarcinelli-Filho et al., 2002; Dev et al., 1997; Caldeira et al., 2007)).

Ya en su parte final, el artículo trata de una aplicación típica en navegación de robots móviles, que es la recuperación del auto-movimiento del vehículo, utilizando el algoritmo propuesto para obtener el campo de flujo óptico correspondiente a las imágenes tomadas por un vehículo uniciclo que dispone de una cámara a bordo y gira sobre su propio eje, y a partir de dicho

campo de flujo óptico recuperar la velocidad de rotación del vehículo. Asimismo se comentan otras posibles situaciones en navegación autónoma en que se podrían utilizar los algoritmos aquí propuestos, para justificar la aplicabilidad de nuestros resultados en este área.

Finalmente, se tiene que resaltar que la estimación del flujo óptico basada en mínimos cuadrados es una técnica diferencial, o sea, utiliza las derivadas espaciales y la derivada temporal de la imagen. Así, para que los vectores estimados correspondan de hecho al movimiento en la imagen, es necesaria una iluminación uniforme y una superficie con reflexión lambertiana. Cuanto más las condiciones de operación se acerquen a estas características ideales, mayor será la correspondencia entre el flujo óptico y el movimiento real. Para garantizar que el algoritmo propuesto se aplicará a imágenes en que dichas condiciones son atendidas, en este artículo se adoptará la secuencia de video *SRI Trees*, frecuentemente utilizada en la literatura de flujo óptico, para prueba de los dos algoritmos propuestos.

2. Base Teórica

2.1. El Campo de Flujo Óptico

El flujo óptico se define como la distribución de las velocidades aparentes de movimiento del patrón de brillo en una imagen. Él se debe al movimiento relativo entre objetos y una cámara, y es registrado siempre que una secuencia de cuadros de imagen es capturada (Horn and Shunck, 1981). La idea básica involucrada en el concepto de flujo óptico es que el brillo de una estructura cualquiera en la imagen permanece constante entre dos cuadros sucesivos. Es decir,

$$I(x; y; t) \approx I(x + dx; y + dy; t + dt), \quad (1)$$

donde x e y son las coordenadas en pixel correspondientes a una estructura en el primero cuadro de imagen (instante t) y $x + dx$ e $y + dy$ son las coordenadas en pixel de la misma estructura en el segundo cuadro de imagen (instante $t + dt$), siendo I el brillo de dicha estructura. De ahí se obtiene, teniendo en cuenta solamente el primer término de la expansión en serie de Taylor de (1),

$$\frac{dI(x; y; t)}{dt} = 0, \quad (2)$$

que es equivalente a

$$I_x u + I_y v + I_t = 0, \quad (3)$$

donde u y v son las componentes del vector de flujo óptico correspondientes al pixel de coordenadas x e y del primer cuadro de imagen (instante t), mientras que I_x , I_y y I_t son las derivadas de la imagen en las direcciones x y y (derivadas espaciales) y en el tiempo (derivada temporal), respectivamente.

La ecuación (3) es denominada la restricción de flujo óptico. A su vez, métodos que utilizan dicha ecuación en el cálculo del flujo óptico, o sea, utilizan las derivadas de la imagen, son llamados métodos diferenciales (basados en gradiente). Sin embargo, dicha ecuación no es suficiente para determinar los valores de u y v en cada pixel de un cuadro de imagen, por presentar

dos incógnitas. Por lo tanto, una restricción adicional debe ser impuesta (Barron and Beauchemin, 1994; Lukas and Kanade, 1981; Horn and Shunck, 1981).

Además de los métodos diferenciales, la literatura referente al cálculo del flujo óptico incluye métodos basados en la coincidencia de regiones, métodos basados en energía, y métodos basados en fase (Barron and Beauchemin, 1994). De todos ellos, sólo el método diferencial de Lukas and Kanade (1981) se tratará en este trabajo, para entenderlo y después modificarlo. Una razón para esto es que este método aún es uno de los más utilizados para calcular el flujo óptico (Carelli et al., 2002; Simas et al., 2010; Caldeira et al., 2007).

2.2. El Algoritmo Clásico de Lukas and Kanade (1981)

Este algoritmo corresponde a un método diferencial, y, por lo tanto, utiliza la ecuación (3) con una restricción de regularización adicional. Dicha restricción es que el vector de flujo óptico es aproximadamente constante en una región de la imagen contenido dos o más píxeles. Básicamente, se considera que cada región cuadrada de N por N píxeles tiene un solo vector de flujo óptico asociado a ella. Así, escribiéndose (3) teniendo en cuenta esta restricción adicional se obtiene

$$I_x^i u + I_y^i v + I_t^i = 0 \quad (4)$$

para $i = 1, \dots, N^2$, donde I_x es la derivada espacial de $I(x; y; t)$ en la dirección x , I_y es la derivada espacial de $I(x; y; t)$ en la dirección y y I_t es la derivada temporal de la imagen, todas en el pixel i . Dichas derivadas son calculadas de la misma forma propuesta en (Horn and Shunck, 1981) y (Caldeira et al., 2007). Con dichas derivadas se escribe un sistema de ecuaciones lineales sobre-determinado en u y v , de lo cual se genera la estimación de mínimo error medio cuadrático (\hat{u}, \hat{v}) del vector de flujo óptico, dada por

$$(\hat{u}, \hat{v}) = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}, \quad (5)$$

siendo $\mathbf{A} = \begin{bmatrix} I_x^1 & \dots & I_x^{N^2} \\ I_y^1 & \dots & I_y^{N^2} \end{bmatrix}^T$ y $\mathbf{b} = -\begin{bmatrix} I_t^1 & \dots & I_t^{N^2} \end{bmatrix}^T$.

A la estimación (\hat{u}, \hat{v}) de flujo óptico así obtenida está asociada la varianza

$$\sigma^2 = \mathbf{e}^T \mathbf{e}, \quad (6)$$

donde $\mathbf{e} = \begin{bmatrix} e^1 & \dots & e^{N^2} \end{bmatrix}^T$, siendo $e^i = I_x^i \hat{u} + I_y^i \hat{v} + I_t^i$ el residuo de (4) en el pixel i , considerando el vector de flujo óptico estimado.

2.3. Determinación de la Estimación Final de un Vector de Flujo Óptico

Dado un conjunto de M estimaciones \hat{u}_i y \hat{v}_i de las componentes \hat{u} y \hat{v} de un vector de flujo óptico, con las respectivas varianzas $\sigma_{\hat{u}_i}^2 = \sigma_{\hat{v}_i}^2$, la estimación óptima en el sentido del error medio cuadrático puede obtenerse a partir de las ecuaciones

$$\frac{\hat{u}}{\sigma_{\hat{u}}^2} = \sum_{i=1}^M \frac{\hat{u}_i}{\sigma_{\hat{u}_i}^2} \quad (7)$$

y

$$\frac{\hat{v}}{\sigma_{\hat{v}}^2} = \sum_{i=1}^M \frac{\hat{v}_i}{\sigma_{\hat{v}_i}^2}. \quad (8)$$

La varianza correspondiente a la estimación óptima es, a su vez,

$$\frac{1}{\sigma_{\hat{u}}^2} = \sum_{i=1}^N \frac{1}{\sigma_{\hat{u}_i}^2}, \quad (9)$$

siendo la misma expresión para \hat{u} y \hat{v} .

3. El Algoritmo Propuesto

El algoritmo que se propone para estimar los vectores de flujo óptico involucra dos pasos. El primero consiste en obtener algunas estimaciones, en base al criterio de mínimos cuadrados, correspondientes al único vector de flujo óptico representante de una región dada de los dos cuadros de imagen bajo análisis, por medio de las ecuaciones (5) y (6). Esto supone que el vector de flujo óptico (\hat{u}, \hat{v}) es constante en esta región, que es la restricción de regularización utilizada en (Lukas and Kanade, 1981). El segundo paso consiste en utilizar dichas estimaciones en las ecuaciones (7), (8) y (9), y así obtener la estimación óptima en el sentido de mínimos cuadrados \hat{u} y \hat{v} del vector de flujo óptico correspondiente a la región bajo análisis.

La implementación de este algoritmo comienza por la división de la imagen bajo análisis en regiones de N por N píxeles, en las cuales se considera el vector de flujo óptico constante. Por lo tanto, sólo un vector de flujo óptico se asocia a cada una de estas regiones (Lukas and Kanade, 1981). La diferencia con respecto al método clásico es que cada región se divide, a su vez, en sub-regiones de N_1 por N_1 píxeles, cada una siendo una pequeña parte de la región considerada (es decir, los vectores de flujo óptico estimados en cada sub-región son diferentes estimaciones del único vector de flujo óptico que caracteriza la región). En el ejemplo presentado adelante se consideran regiones de 20×20 píxeles, cada una dividida en 16 sub-regiones de 5×5 píxeles, como se ilustra en la Figura 1.

Por lo tanto, la aplicación del algoritmo clásico de mínimo error cuadrático (Lukas and Kanade, 1981) en cada sub-región generará 16 estimaciones del vector de flujo óptico de dicha región de 20×20 píxeles, cada una con su varianza asociada. Dichas estimaciones generan la estimación óptima (ecuaciones 7 y 8) del vector de

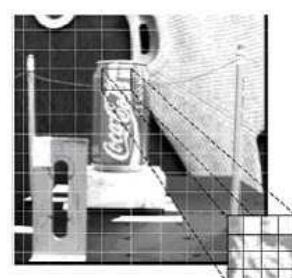


Figura 1: Regiones y sub-regiones utilizadas por el algoritmo propuesto.

flujo óptico asociado a la región considerada (téngase en cuenta que el representante de los pixeles de la sub-región es el pixel central, que siempre existe cuando se elige un número impar de pixeles para cada sub-región - 5x5, por ejemplo).

Así, considerando regiones de $N \times N$ pixeles y sub-regiones de $N_1 \times N_1$ pixeles, el esfuerzo de cómputo necesario para obtener el vector de flujo óptico asociado a cada región del cuadro de imagen incluye el cálculo de las derivadas espaciales y temporal de la imagen (I_x , I_y y I_t) en N^2 pixeles, $\frac{N^2}{N_1^2}$ veces el cálculo de las ecuaciones (5) y (6), y una vez el cálculo de las ecuaciones (7), (8) y (9). Por lo tanto, este nuevo algoritmo demanda algún tiempo más de cómputo, aunque no mucho más comparado al algoritmo de Lukas y Kanade (1981), debido a la adición del paso de cálculo de la estima óptima. La Tabla 1 presenta, en sus columnas 2 y 3, respectivamente, el esfuerzo de cómputo asociado al algoritmo original de Lukas and Kanade (1981) y al algoritmo modificado propuesto (que se nombrará, de aquí en adelante, algoritmo de mínimos cuadrados con fusión - AMCF).

Para validar la afirmación de que la estima final del vector de flujo óptico correspondiente a una región de la imagen bajo análisis es más precisa, se presenta en la Tabla 2 las componentes horizontales (\hat{u}) de las dieciséis estimas del vector de flujo óptico correspondiente a una región de 20×20 pixeles de la Figura 1, con sus respectivas varianzas, obtenidas mediante la aplicación de las ecuaciones (5) y (6) a sub-regiones de 5×5 pixeles de dicha región. En dicha tabla se tiene una fila con valores 0 y 125 cuando la matriz $\mathbf{A}^T \mathbf{A}$ en (5) no es invertible, no permitiendo así la obtención de \hat{u} para la sub-región correspondiente. En este caso, se atribuyen los valores 0 y 125 a \hat{u} y su varianza, respectivamente. Los valores en la tabla son entonces utilizados en las ecuaciones (7) y (9), resultando en la estima óptima presentada en la Tabla 3. El resultado es que el paso de fusión de datos introducido en el algoritmo AMCF de hecho mejora la estima final del vector de flujo óptico, en el sentido de que la varianza correspondiente es sensiblemente más pequeña que la más pequeña de las varianzas asociadas a las dieciséis estimas iniciales.

Entonces, como se puede observar en la Tabla 3, el algoritmo AMCF difiere del algoritmo original de Lukas and Kanade (1981) por el paso de fusión de datos introducido, el cual garantiza que el nuevo algoritmo propuesto tiene un desempeño superior al algoritmo original. Por lo tanto, tras analizar las tablas presentadas, uno puede concluir que el nuevo algoritmo AMCF es atractivo para aplicaciones en que se necesita una estima de flujo óptico más precisa.

4. Modificación del Algoritmo Propuesto para Reducir su Coste Computacional

Aunque sea atractivo por producir estimas más precisas de los vectores de flujo óptico, el algoritmo descrito en la sección anterior consume más tiempo de cálculo que lo necesario, cuando se trata de aplicaciones en que no se requiere obtener estimas precisas de dichos vectores. Un buen ejemplo es el problema de evitar obstáculos en la navegación de robots móviles (Sarcinelli-Filho et al., 2002; Dev et al., 1997; Caldeira et al.,

Tabla 1: Esfuerzo de cómputo para los algoritmos analizados.

Algoritmos			
Cálculo	Lukas and Kanade (1981)	AMCF	AMCFM
I_x	N^2	N^2	$2N^2/N_1^2$
I_y	N^2	N^2	$2N^2/N_1^2$
I_t	N^2	N^2	$2N^2/N_1^2$
Ec. 5	1	N^2/N_1^2	N^2/N_1^2
Ec. 6	1	N^2/N_1^2	N^2/N_1^2
Ec. 4	1	N^2	$2N^2/N_1^2$
Ec. 7	0	1	1
Ec. 8	0	1	1
Ec. 9	0	1	1

Tabla 2: Estimas iniciales para la componente horizontal del vector de flujo óptico, y sus varianzas.

Valor	Varianza
0.2834	1.6591
0.2026	2.7444
0.1527	2.6940
0.0000	125.0000
0.2093	6.1890
0.1634	1.3950
0.0716	2.1727
0.0000	125.0000
0.1023	1.4112
0.1479	5.2418
-0.3947	5.0480
0.0000	125.0000
0.1213	0.8270
-0.2922	4.0241
0.2215	3.4557
0.0000	125.0000

2007). En este caso, el robot puede desviarse de los obstáculos aún cuando la estima de su distancia al obstáculo, obtenida de la estima del tiempo a choque que se obtiene de las coordenadas del foco de expansión del campo de flujo óptico (Caldeira et al., 2007), contiene gran incertidumbre. Entonces, puede ser muy interesante disponer de un algoritmo de cálculo de los vectores de flujo óptico que sea más rápido, aunque produzca estimas más groseras de los vectores de flujo óptico. De hecho, la reducción del tiempo de cálculo en esta situación puede ser imprescindible para permitir el cálculo en tiempo real con la capacidad de procesamiento comúnmente disponible a bordo y con el robot moviéndose a media/alta velocidad.

Para obtener un algoritmo más rápido, se modifica el modo como se obtienen las estimas del vector de flujo óptico correspondientes a las sub-regiones, en la primera etapa del algoritmo AMCF. En lugar de tomar las derivadas de la imagen en

Tabla 3: Estima óptima correspondiente a los datos en la Tabla 2.

Valor	Varianza
0.1163	0.1801

Tabla 4: Esfuerzo de cómputo para los algoritmos analizados, para regiones de 20x20 pixeles y sub-regiones de 5x5 pixeles.

Cálculo	Algoritmos		
	Lukas and Kanade (1981)	AMCF	AMCFM
I_x	400	400	32
I_y	400	400	32
I_t	400	400	32
Ec. 5	1	16	16
Ec. 6	1	16	16
Ec. 4	400	400	32
Ec. 7	0	1	1
Ec. 8	0	1	1
Ec. 9	0	1	1

todos los $N_1 \times N_1$ pixeles de cada sub-región, solamente dos de ellos son elegidos (el número mínimo de pixeles necesarios para obtener \hat{u} y \hat{v}), similar a lo que se hace en (Caldeira et al., 2007). De este modo, en vez de calcular las derivadas de la imagen en todos los $N \times N$ pixeles pertenecientes a cada región, se las calculan solamente en $(N \times N)/(N_1 \times N_1)$ pixeles. Detalles del esfuerzo de cómputo correspondiente a dicho algoritmo se muestran en la Tabla 1, en la columna del extremo derecho. El resultado de dicha estrategia es que cada una de las estimas iniciales del vector de flujo óptico será menos precisa debido al bajo número de observaciones, pero habrá una reducción significativa en el tiempo de cálculo. Por lo tanto, este algoritmo, a continuación denominado algoritmo de mínimos cuadrados con fusión modificado (AMCFM), representa una solución de compromiso, requiriendo menos esfuerzo computacional para generar estimas iniciales menos precisas para cada vector de flujo óptico. Sin embargo este algoritmo es muy atractivo para aplicaciones que no requieren elevada precisión en la estima de los vectores de flujo óptico.

Como ejemplo, a partir de la Tabla 1 se puede comparar el esfuerzo de cómputo correspondiente a los algoritmos AMCF, AMCFM y el algoritmo de Lukas and Kanade (1981) cuando las regiones son de 20x20 pixeles y las sub-regiones son de 5x5 pixeles, lo que resulta en los datos presentados en la Tabla 4, de la cual se puede notar la reducción del esfuerzo de cómputo cuando se adopta el algoritmo AMCFM.

Como se ha visto, este artículo propone dos versiones modificadas del algoritmo clásico de mínimos cuadrados (Lukas and Kanade, 1981), una que estima de forma más precisa los vectores flujo óptico y requiere más esfuerzo de cómputo (AMCF), y otra que estima de forma más grosera los vectores de flujo óptico, pero con menos esfuerzo de cómputo (AMCFM). El usuario puede, entonces, elegir la versión más adecuada para su aplicación.

5. Validación de los Algoritmos Propuestos

A fines de probar el buen desempeño del algoritmo de mínimos cuadrados con fusión, se utilizaron los dos primeros cuadros de la secuencia de video *SRI Trees* (el primero de ellos está en la Figura 2) para calcular los vectores de flujo óptico

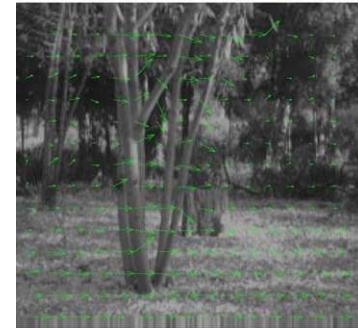


Figura 2: Primero cuadro de imagen de la secuencia de video *SRI Trees*, con los vectores de flujo óptico correspondientes.

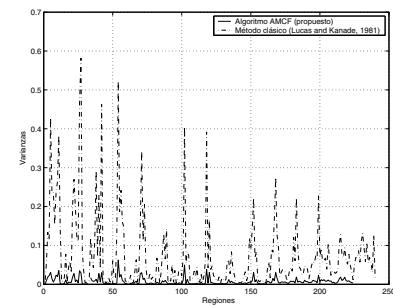


Figura 3: Varianzas asociadas a los vectores de flujo óptico estimados utilizando el algoritmo de mínimos cuadrados clásico (Lukas and Kanade, 1981) y el algoritmo de mínimos cuadrados con fusión (AMCF), considerando regiones de 20x20 pixeles de la Figura 2.

correspondientes a regiones de 20x20 pixeles. La Figura 3 presenta una gráfica conteniendo las varianzas correspondientes a los vectores de flujo óptico estimados en las regiones de la Figura 2 cuando se utiliza el algoritmo clásico de Lukas and Kanade (1981) y el de mínimos cuadrados con fusión. Como se puede observar, las estimas obtenidas con el algoritmo de mínimos cuadrados con fusión aquí propuesto son notablemente más precisas.

A continuación, para comparar la precisión de los vectores de flujo óptico estimados con los dos algoritmos propuestos (AMCF y AMCFM), la Figura 4 presenta las varianzas obtenidas cuando se utilizan dichos algoritmos para calcular los vectores de flujo óptico correspondientes al mismo ejemplo. En dicha figura se puede percibir claramente que las varianzas correspondientes al algoritmo AMCF son más pequeñas, aunque el algoritmo AMCFM también genera estimas con varianza baja, para algunas regiones.

También en referencia al desempeño de los algoritmos AMCF y AMCFM, se realizaron las mismas pruebas utilizando otras tres secuencias de video estándar en la literatura de flujo óptico, la *Rubik Cube*, la *NASA* y la *Hamburg Taxi*, y los resultados son similares a los que se han presentado en esta sección.

Finalmente se debe destacar que el tamaño elegido para cada región analizada en cada cuadro de imagen en los ejemplos mostrados, es decir, 20x20 pixeles, no es rígido, pudiéndose

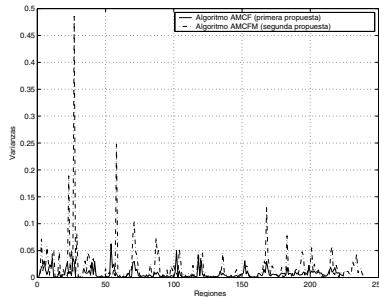


Figura 4: Varianzas asociadas a los vectores de flujo óptico obtenidos con el algoritmo AMCF y con el algoritmo AMCFM, para el mismo ejemplo de la Figura 3.

elegir otros tamaños para dichas regiones (teniendo en cuenta las características específicas de la aplicación). Sin embargo, cuanto menor la región menos estimas iniciales estarán disponibles para generar la estimación óptima. Por ejemplo, si se seleccionaron regiones con 9×9 pixeles, con sub-regiones de 3×3 pixeles, se tendrán disponibles 9 estimas iniciales para generar la estimación óptima para cada vector de flujo óptico. En este caso, el número de pixeles en el cálculo de las derivadas de la imagen sería 81 por cada región, si se utiliza el algoritmo AMCF, o 18 por cada región, si se utiliza el algoritmo AMCFM. En otras palabras, el esfuerzo de cómputo es menor cuando se definen regiones (y sub-regiones) con menos pixeles, pero el número de las estimas iniciales del vector de flujo óptico disponibles para el paso de fusión es menor. Otra característica importante es que el cálculo de las derivadas de la imagen se hace de la forma menos compleja propuesta en la literatura (Horn and Shunck, 1981; Caldeira et al., 2007).

6. Aplicación de los Algoritmos Propuestos en la Navegación de Robots Móviles

En esta sección son comentadas algunas posibles aplicaciones del algoritmo AMCFM en la navegación de robots móviles. Dichas aplicaciones no son analizadas en detalles, lo que escapa al objetivo del trabajo, pero sí comentadas brevemente, con el fin de sugerir posibles continuaciones de este trabajo.

La primera de estas aplicaciones es una modificación de la estrategia propuesta en (de Micheli and Verri, 1993), para recuperar el movimiento de rotación del robot móvil durante su navegación, así como el tiempo al choque con un obstáculo (Caldeira et al., 2007), con base en flujo óptico de una dimensión (en este caso, la componente horizontal y la componente vertical del vector de flujo óptico, respectivamente, calculadas en una línea vertical pasando por el centro de la imagen y en una línea horizontal cuya ordenada es fija y distinta de cero, respectivamente, teniendo en cuenta el plano de la imagen). Con este fin, se considera que el robot móvil está moviéndose en un plano horizontal, con rotación sólo alrededor del eje Z y traslaciones sólo en los ejes X e Y. En este caso, como en (de Micheli and Verri, 1993), los errores Δu y Δv en el cálculo de los componentes horizontal y vertical, respectivamente, del vector de

flujo óptico están relacionados con el error $\Delta\Omega$ en la velocidad angular del robot móvil a través de la ecuación

$$\Delta\Omega = \frac{1}{f} \left(\Delta u + \left| \frac{x}{y} \right| \Delta v \right), \quad (10)$$

donde f es la longitud focal de la cámara utilizada. Por lo tanto, el error será mínimo en los puntos donde $x = 0$, es decir, los puntos que están en la línea vertical l_v a través del centro de proyección (centro óptico), que en este caso es el centro de la imagen. Por lo tanto, haciendo $x = 0$ se obtiene (de Micheli and Verri, 1993)

$$\Omega = \frac{u}{f}, \quad (11)$$

donde se percibe que u es constante sobre l_v . Por lo tanto, utilizando (11) se puede recuperar la velocidad angular del robot móvil. Para esto, se computan diversas estimas de la componente horizontal u del flujo óptico a lo largo de la línea vertical l_v , estimándose Ω como

$$\hat{\Omega} = \frac{\hat{u}}{f}, \quad (12)$$

tomándose \hat{u} como el resultado de la estimación óptima obtenida a partir de todos los valores de u tomados en l_v . En otras palabras, se puede estimar \hat{u} utilizando el algoritmo de mínimos cuadrados con fusión, o bien su versión modificada, aquí propuestos, utilizando, por ejemplo, sub-regiones de 5×1 pixeles a lo largo de l_v . Tenga en cuenta que se trata de una metodología diferente a la propuesta por de Micheli and Verri (1993), que considera el promedio de los valores de u tomados a lo largo de l_v . Es importante destacar, entonces, que esta propuesta genera un valor de Ω más preciso, ya que la estimación óptima de \hat{u} es más precisa que la media.

Como ejemplo de esta aplicación, la Figura 5 muestra los valores de la velocidad angular que se recuperó por medio de la estrategia mencionada, teniendo en cuenta una situación en la que se dispara un robot móvil (el Pioneer 3-DX, de *Mobile Robotics, Inc.*) con una velocidad lineal nula y una velocidad angular constante de $10^\circ/s$. Durante este movimiento una secuencia de imágenes fue capturada, y para cada par de imágenes consecutivas se estima la componente horizontal del flujo óptico, teniendo en cuenta la línea l_v , donde el flujo óptico es constante, dividiéndola en sub-regiones de 5×1 pixeles y utilizando el algoritmo de mínimos cuadrados con fusión, para determinar diferentes estimas de u . A partir de estos valores de u se genera la estimación \hat{u} como se propone en la sección 3. Las velocidades angulares estimadas a través de la ecuación (12) se muestran en la Figura 5. En la misma figura también se muestran los valores de velocidad angular obtenidos a partir de la odometría del robot, para comparación. Como puede verse, los valores de la velocidad angular obtenidos por el método propuesto aquí son muy próximos a los obtenidos por odometría, y están muy cerca de la velocidad angular nominal del vehículo, confirmando la validez de la metodología aquí sugerida para recuperar la rotación del robot.

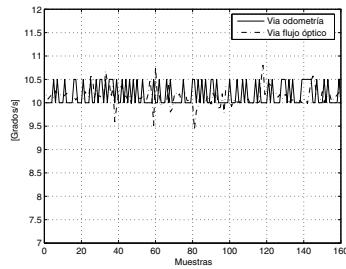


Figura 5: Velocidad angular del robot obtenida a través del uso de su odometría y de la metodología aquí propuesta, basada en el cálculo del flujo óptico.

Teniéndose el valor de la velocidad angular, también se puede recuperar el valor $\frac{1}{\tau}$, que caracteriza la estructura de la escena (τ es conocido como el tiempo al choque, y corresponde al tiempo para que el centro de proyección de la cámara a bordo del robot alcance el foco de expansión del campo de flujo óptico (de Micheli and Verri, 1993; Caldeira et al., 2007), manteniéndose su velocidad lineal actual). Por definición, el foco de expansión del campo de flujo óptico es el punto donde todos los vectores de flujo óptico salen, cuando la cámara se mueve en la dirección de un objeto dado, que en el caso de navegación robótica sería un obstáculo. De acuerdo con de Micheli and Verri (1993), una vez recuperada la velocidad angular Ω se puede determinar el valor del tiempo al choque como,

$$\frac{1}{\tau} = \frac{v}{y} - \frac{x}{f}\Omega. \quad (13)$$

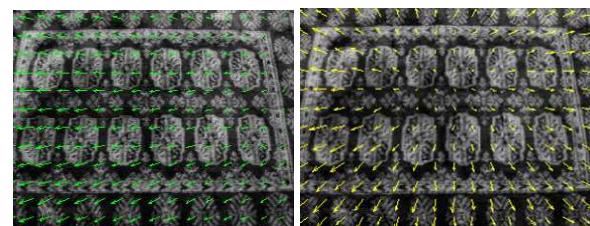
De este modo se puede calcular el tiempo al choque a partir de la componente vertical del flujo óptico computada a lo largo de una línea horizontal l_h , definida por un valor y constante no nulo, tan grande como sea posible. Téngase en cuenta que el valor τ debe ser constante para dos cuadros de imagen. Para esto, como el valor x varía a lo largo de l_h , es necesario eliminar el efecto de Ω , lo que equivale a decir que el robot debe tener un movimiento solo de traslación, a fin de lograr obtener una buena estima del tiempo al choque (Caldeira et al., 2007).

Una forma de hacer esto es restar el valor de Ω , estimado como se ha sugerido antes, de las componentes horizontales de todos los vectores de flujo óptico correspondientes a un cuadro dado de imagen. Haciendo esto, el nuevo campo de flujo óptico corresponderá solamente a la traslación del robot móvil, y a partir de él se podrá determinar el tiempo a choque utilizando la ecuación (12), ahora con $\Omega = 0$. Las Figuras 6(a) y 6(b) ilustran esta idea: en la Figura 6(a) se tiene el campo de flujo óptico correspondiente al movimiento de rotación añadido al movimiento de traslación, mientras que en la Figura 6(b) se tiene el campo de flujo óptico corregido restando la rotación Ω de la componente horizontal de cada vector de flujo óptico. Como se puede percibir en dicha figura, el nuevo campo de flujo óptico obtenido tiene un foco de expansión muy bien definido, coincidente con el objeto observado por la cámara, lo que permite caracterizar muy bien el tiempo al choque.

Otra posible aplicación de los algoritmos aquí propuestos

para cálculo del flujo óptico sería una situación en que un vehículo aéreo cargando una cámara tiene que seguir a otro vehículo en tierra. En este caso, conocidas las características del vehículo a ser seguido, por ejemplo el color, se puede segmentar de las imágenes que lo contienen, eligiendo así la región de la imagen que corresponde al mismo. Entonces se define una región del plano de la imagen que se mueve con velocidad constante, es decir, que corresponde al vehículo que se desea seguir, y así se caracteriza un solo vector de flujo óptico asociado al movimiento del vehículo con respecto al observador (cámara a bordo del vehículo aéreo). Luego se puede definir sub-regiones pertenecientes a la región que corresponde al vehículo a ser seguido, y aplicar los algoritmos aquí propuestos para determinar el flujo óptico y la velocidad relativa vehículo-cámara.

Finalmente, otra posibilidad de utilización de los algoritmos aquí propuestos sería la situación de control servo-visual en que un robot móvil con una cámara a bordo tiene que navegar por un pasillo. Por seguridad, es adecuado que el robot se mantenga al medio del pasillo si no hay obstáculos. En este caso, se definen dos regiones de interés, que son dos regiones simétricas definidas en las laterales del pasillo, en las cuales se define un solo componente horizontal del vector de flujo óptico. De esta forma, un controlador basado en la disparidad entre las dos componentes horizontales así definidas controla la rotación del robot móvil, de manera de hacer cero la disparidad entre los valores absolutos de las componentes horizontales de los dos vectores de flujo óptico, lo que corresponderá al robot móvil navegando por el medio del pasillo. Teniéndose en cuenta que las componentes horizontales de los vectores de flujo óptico en cada una de dichas regiones son constantes, los algoritmos aquí propuestos pueden ser aplicados al problema. De este modo se calcula solo la componente horizontal correspondiente al flujo óptico utilizando uno de los algoritmos aquí propuestos, y considerando cada una de las regiones de interés en cada cuadro de imagen.



(a) Campo de flujo óptico para rotación combinada con traslación. (b) Campo de flujo óptico después de compensada la rotación.

Figura 6: Compensación del movimiento de rotación incluido en el campo de flujo óptico, para definición del foco de expansión.

7. Conclusiones

Este trabajo explora el uso del paradigma de la fusión de datos, con base en la estimación óptima en el sentido de mínimos cuadrados, para obtener una mejor estima del vector de flujo óptico correspondiente a una región de un cuadro de imagen. Teniendo en cuenta un algoritmo de mínimos cuadrados clásico en la literatura, se propone un nuevo algoritmo para el

cálculo de los vectores de flujo óptico y se consideran dos implementaciones de este algoritmo. La primera implementación corresponde simplemente a añadir el paso de estima óptima, lo que resulta en una alta precisión de la estimación y en el crecimiento del tiempo de cómputo. La segunda implementación corresponde a agregar el paso de estima óptima después de reducir el número de pixeles utilizados para obtener las estimas iniciales. Este algoritmo genera estimas menos precisas de los vectores de flujo óptico, debido a que las estimas iniciales son menos fiables, pero permite una reducción significativa del tiempo de cálculo. En aplicaciones donde se puede prescindir de los cálculos de alta precisión de los vectores de flujo óptico, tales como navegación de robots móviles, el menor esfuerzo computacional hace que este segundo algoritmo se torne atractivo, mientras que el primero es interesante para aplicaciones donde una alta precisión de la estima es fundamental.

Después de validar los algoritmos propuestos a través de ejemplos y análisis teórico, el artículo también sugiere situaciones en el contexto de navegación de robots móviles en que es factible su aplicación.

Por lo tanto, como lo demuestra el análisis teórico y los resultados experimentales, los algoritmos propuestos son viables, y su uso en aplicaciones tales como navegación de robots móviles es muy atractivo, sobre todo cuando se elige el segundo algoritmo, el más rápido en su cálculo.

English Summary

Adding a Data Fusion Step to the Least-Mean-Square-Error Algorithm to Improve Optical Flow Field Estimation

Abstract

A new approach for optical flow estimation is here addressed, in which some least-mean-square-error estimates of an optical flow vector are fused to generate a more accurate estimate of such vector. How to get the initial least-mean-square-error estimates is also discussed, looking for a faster algorithm. Results of using the proposed method to estimate the optical flow vectors for two frames of the well known *SRI Trees* video sequence are presented and discussed. Finally, some situations in the context of mobile robot navigation, in which the algorithms proposed here could be applied, are mentioned, to highlight the applicability of the results of this research.

Keywords:

Optical flow, Data fusion, Optimal estimation, Least-squares estimation, Autonomous mobile robots.

Agradecimientos

Los autores agradecen a la Secretaría de Políticas Universitarias del Ministerio de la Educación, Ciencia y Técnica de la República Argentina y a la CAPES - Fundación Coordinación de Perfeccionamiento del Personal de Enseñanza Superior, del

Ministerio de la Educación de la República Federativa del Brasil, por el soporte a la colaboración entre la Universidad Nacional de San Juan, San Juan, Argentina, y la Universidad Federal del Espíritu Santo, Vitoria, ES, Brasil, en cuyo marco se ha desarrollado este trabajo.

Referencias

Barron, J. L., Beauchemin, S. S., 1994. Performance of optical flow techniques. *International Journal of Computer Vision* 12, 43–77.

Black, M. J., Anandan, P., 1990. A model for the detection of motion over time. In: *Proceedings of the International Conference on Computer Vision*. Osaka, Japan, pp. 33–37.

Caldeira, E. M. O., Schneebeli, H. J. A., Sarcinelli-Filho, M., 2007. An optical flow-based sensing system for reactive mobile robot navigation. *SBA Controle e Automação* 18 (3), 265–277.

Carelli, R., Soria, C. M., Nasini, O. H., Freire, E. O., 2002. Stable agv corridor navigation with fused vision-based control signals. In: *Proceedings of the 28th Annual Conference of IEEE Industry Electronics Society - IECON 02*. Sevilla, Spain, pp. 2433–2438.

de Micheli, E., Verri, A., 1993. Vehicle guidance from one-dimensional optical flow. In: *Proceedings of the 1993 IEEE Intelligent Vehicle Symposium*. Tokio, Japan, pp. 183–188.

Dev, A., Krose, B. J. A., Groen, F. C. A., 1997. Navigation of a mobile robot on the temporal development of the optic flow. In: *Proceedings of the 1997 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems - IROS 97*. Grenoble, France, pp. 558–563.

Gern, A., Moebus, R., Franke, U., 2002. Vision-based lane recognition under adverse weather conditions using optical flow. In: *Proceedings of the 2002 IEEE Intelligent Vehicle Symposium*, pp. 652–657.

Giachetti, A., Campani, M., Torre, V., 1998. The use of optical flow for road navigation. *IEEE Transactions on Robotics and Automation* 14, 34–48.

Hong, L., 1999. Sense your world better: multi-sensor/information fusion. *IEEE Journal of Circuits and Systems* 10, 7–8, 12–15, 28.

Horn, B. K. P., Shunck, B. G., 1981. Determining optical flow. *Artificial Intelligence* 17, 185–204.

Jia, Z., Balasuriya, A., Challa, S., 2008. Vision based data fusion for autonomous vehicles target tracking using interacting multiple dynamic models. *Computer Vision and Image Understanding* 109, 1–21.

Langley, K., Atherton, T. J., Wilson, R. G., Lacombe, M. H. E., 1991. Vertical and horizontal disparities from phase. *Image and Vision Computing* 9, 296–302.

Lukas, B., Kanade, T., 1981. An iterative image registration technique with an application to stereo vision. In: *Proceedings of the 7th International Joint Conference on Artificial Intelligence - IJCAI 81*. Vancouver, Canada, pp. 674–679.

Prince, J. L., McVeigh, E. R., 1992. Motion estimation from tagged mr image sequences. *International Journal of Computer Vision* 11, 238–249.

Rudall, B. H., 2004. Reports and surveys. *Robotica* 22, 1–8.

Sarcinelli-Filho, M., Schneebeli, H. A., Caldeira, E. M. O., 2002. Using optical flow to control mobile robot navigation. In: *Proceedings of the 15th IFAC World Congress on Automatic Control*. Barcelona, Spain.

Simas, G. M., Fickel, G. P., Novelo, L., de Bem, R. A., Botelho, S. S. C., 2010. Reconstrução probabilística 3d aplicada à análise de movimento: um enfoque na teleoperação. In: *Anais do XVII Congresso Brasileiro de Automática - CBA 2010*. Bonito, MS, Brasil, pp. 3800–3807.

Watanabe, Y., Fabiani, P., Besnerais, G. L., 2009. Simultaneous visual target tracking and navigation in a gps-denied environment. In: *Proceedings of the 14th International Conference on Advanced Robotics - ICAR 09*. Munich, Germany.

Zainal, Z., Ramli, R., Mustafa, M. M., 2012. Improved heading direction interpretation via optical flow using selected region of interest. *Procedia Engineering* 41, 1298 – 1306.

Zhang, Z., Faugeras, O., 1992. Three-dimensional motion computation and object segmentation in a long sequence in stereo frames. *International Journal of Computer Vision* 7, 211–241.

Zheng, Q., Chellappa, R., 1993. Automatic feature point extraction and tracking in image sequences for unknown image motion. In: *Proceedings of the International Conference on Computer Vision*. Berlin, Germany, pp. 335–339.