

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 709 211**

51 Int. Cl.:

G06T 7/20 (2007.01)

G11B 27/00 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **25.05.2016** E 16382233 (1)

97 Fecha y número de publicación de la concesión europea: **26.09.2018** EP 3249609

54 Título: **Método implementado por ordenador y sistema para procesamiento de video con consistencia en el tiempo**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
15.04.2019

73 Titular/es:

RATIONAL PIXELS S.L. (100.0%)
C/ Provença 397, P 2º 2a
08025 Barcelona, ES

72 Inventor/es:

GARAMENDI, JUAN FRANCISCO;
ARIAS MARTÍNEZ, PABLO y
SADEK SADEK, RIDA

74 Agente/Representante:

CARVAJAL Y URQUIJO, Isabel

ES 2 709 211 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín Europeo de Patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre Concesión de Patentes Europeas).

DESCRIPCIÓN

Método implementado por ordenador y sistema para procesamiento de video con consistencia en el tiempo

Campo de la invención

5 La presente invención se comprende dentro de los campos del procesamiento de video y edición de video, mediante la cual un video dado se modifica para lograr un cierto objetivo.

Antecedentes de la invención

10 La edición digital de secuencias de video se ha convertido en una etapa común para la post producción de cine y video, principalmente debido a los avances en los campos de gráficos por ordenador y visión artificial. Las tareas de edición de video varían desde las operaciones básicas tales como arreglos, recortes, particiones y cambios de tamaño de segmentos de video a los más elaborados tales como filtros de efectos, edición de texturas de objetos, eliminación y adición de objetos en un segmento de video, entre otros.

15 Una diferencia significativa entre la edición de video y la edición de imagen (fija o fotografía) es el requisito de que el resultado tenga que ser coherente a lo largo del tiempo. La coherencia temporal se define como una transición suave entre los fotogramas sucesivos, coherentes con el movimiento de los objetos en la secuencia. Debido a esta restricción, la edición de un video no se puede reducir a una serie de problemas de edición de imágenes independientes. La dependencia temporal impuesta por el movimiento tiene que tenerse en cuenta.

20 Diversas aproximaciones al problema de edición de video estiman las trayectorias a partir del video, y calculan el video editado como el mínimo de un funcional de energía. En este contexto, el video, o una región de interés (ROI) en él, se representa como un vector en \mathbb{R}^N donde el número de variables N corresponde al número de píxeles en el ROI. Por ejemplo para un ROI rectangular de ancho W , alto H , T fotogramas y color codificado usando un espacio de color de 3 canales, por ejemplo, RGB, se tiene $N = 3WHT$. El video editado es calculado mediante la minimización de una funcional de energía $E : \mathbb{R}^N \rightarrow \mathbb{R}$ con una técnica de optimización adecuada. El funcional de energía se diseña buscando que sus mínimos correspondientes tengan las "propiedades adecuadas" de la solución buscada. Estas propiedades vienen dadas por la tarea de edición específica y ciertas propiedades generales, tales como coherencia espacial y temporal.

25 En particular, se enfoca en funcionales de energía de video que tengan la siguiente estructura:

$$E(u) = \sum_{t=0}^T E_t^e(u_t) + \sum_{t=0}^{T-1} E_{t,t+1}^{tc}(u_t, u_{t+1}) \quad (\text{Ecuación 1})$$

30 Donde $u \in \mathbb{R}^N$ indica el video desconocido vectorizado, $t = 0, \dots, T$ es el índice de fotograma, y u_t representa el t -ésimo fotograma de u (también como un vector). La ecuación (Ecuación 1) establece que la energía E se puede descomponer como una suma de dos tipos de términos.

Los términos en el primer sumatorio consisten en energías $E_t^e(u_t)$ que afectan a un solo fotograma. Su forma específica depende de las tareas de edición. Por ejemplo, funcionales como el siguiente se han usado a menudo en la literatura:

$$E_t^e(u_t) = \sum_{x \in \Omega} (u(x, t) - f(x, t))^2 + \frac{\lambda}{p} \sum_{x \in \Omega} |\nabla u(x, t) - g(x, t)|^p. \quad \text{Ecuación 1.1}$$

35 Donde Ω denota el dominio del fotograma (típicamente un rectángulo), $x \in \Omega$ son las coordenadas de píxel, de manera que, $u(x, t)$ es el nivel de gris (color) del píxel localizado en x del fotograma t del video u . ∇ es el operador de gradiente espacial discreto (por ejemplo usando diferencias finitas), $\lambda, p \in R$ son parámetros del funcional, f es un video dado y g es un campo vectorial (por ejemplo el gradiente espacial de un video); f y g son conocidos, típicamente como el resultado de una etapa de procesamiento previa. El primer sumatorio es un término de fidelidad de datos al video f dado y el segundo sumatorio establece un acoplamiento del gradiente de la solución con g , medido con la norma- p . Como ejemplo, si f es el video original y $g = 0$, se puede diseñar un filtro de suavizado. Si $p = 2$ el suavizado resultante es equivalente a un desenfoque gaussiano del video original. Si $p = 1$, el suavizado preserva los bordes. Otro ejemplo podría ser un filtro para separar textura de la componente geométrica, es decir, obtener una versión de un video como si fuera "dibujos animados". Esto se lograría definiendo g como versión simplificada del gradiente del video original, manteniendo solo grandes gradientes (asociados a la información geométrica definida por bordes fuertes) y eliminando los gradientes pequeños (asociados con la textura, los detalles, etc.). Estos ejemplos sólo son dados aquí para fijar las ideas. La forma específica del término E_t^e de energía depende del tipo de edición deseada y problema a resolver, y puede no tener la estructura dada en la Ecuación 1.1, excepto por el hecho de que solo depende del fotograma t .

Los términos en el segundo sumatorio $E_{t,t+1}^{tc}(u_t, u_{t+1})$ acoplan pares de fotogramas contiguos. Su objetivo es imponer la coherencia temporal penalizando alguna medida de la variación del video a lo largo de una trayectoria de movimiento. La forma específica de los términos de coherencia temporal $E_{t,t+1}^{tc}(u_t, u_{t+1})$ depende de la elección del criterio de coherencia temporal. Se darán ejemplos mas adelante.

- 5 Sin el acople temporal impuesto por los términos de coherencia temporal $E_{t,t+1}^{tc}(u_t, u_{t+1})$, la minimización de la energía resultante se reduce a las minimizaciones de los términos $E_t^e(u_t)$ que afectan a un funcional. Cada una de estas se puede minimizar independientemente de cada fotograma. A la vez que esto es atractivo desde un punto de vista computacional ya que permite la paralelización, no hay garantía de que el resultado sea consistente en el tiempo. Por el contrario, los términos de coherencia temporal acoplarán todos los pares de
- 10 fotogramas adyacentes, lo cual implica que la energía tiene que ser minimizada simultáneamente sobre la totalidad del volumen del video. Esto impide la paralelización sobre los fotogramas. Además, a menudo ocurre el caso donde los costes computacionales de minimizar una energía conjuntamente sobre T fotogramas, son mucho más altos que T veces el coste de minimizar la energía sobre un solo fotograma (La complejidad del algoritmo de minimización se escala de forma superlineal o exponencialmente con el número de variables).

15 Revisión de edición de video con coherencia temporal

Los métodos de edición de video con coherencia temporal se pueden clasificar de acuerdo con el patrón de movimiento usado. La gran mayoría de software de edición de video profesional se basa en patrones de movimiento definidos de manera cerrada mediante parámetros. Los modelos paramétricos trabajan con suposiciones hechas sobre la geometría de la escena. El caso más común es suponer que la escena es plana

20 (no curva) a trozos [25,14]. En la post producción de cine y video profesional, hay diversos programas de software que le permiten a un artista de efectos visuales, seleccionar una región plana la cual es entonces seguida de manera automática por el software. Ejemplos de ellos son *mocha* [22], o seguimiento de plano de *Nuke* [11]. Este patrón permite el cálculo de un mapeado simple entre cualquier par de fotogramas, dicho mapeado se puede usar entonces para propagar la información de un fotograma a otro. Cuando un objeto en

25 el dominio de edición no es plano (es curvo), el artista necesita segmentarlo en piezas que pueden ser aproximadas por un plano, y realizar un seguimiento para cada region plana. Este proceso toma tiempo y a menudo el resultado necesita retoques para eliminar cualquier marca entre los diferentes rastreadores.

Por otra parte, los patrones no paramétricos no hacen suposiciones sobre la geometría de la escena. Estos patrones usualmente calculan el movimiento en la secuencia usando flujo óptico. En los años recientes ha

30 habido un progreso considerable en el cálculo del flujo óptico. Por ejemplo, los algoritmos de flujo óptico del estado de la técnica son capaces de manejar algunos desplazamientos grandes y permiten fuertes discontinuidades en el movimiento. Este es el caso de [21,8,6,2] por citar unos pocos. Estos métodos sufren aún del problema de “apertura”: el componente del vector de movimiento que es tangente a las curvas de nivel de la imagen no se puede calcular. En la práctica, para aliviar este problema, se incorpora un término de

35 regularidad. El término de suavidad ocasiona un efecto de llenado que provoca un cálculo del flujo denso, incluso aunque esté presente el problema de la apertura.

A continuación, se revisa el estado de la técnica de la edición de video con coherencia temporal basado en el flujo óptico. Aunque se han usado diversos efectos de flujo óptico en la post producción [19] de cine y video profesionales, su uso para edición de video consistente en el tiempo es aún marginal en comparación con el

40 uso extendido de seguimiento de regiones planas.

Ejemplos de términos de energía con coherencia en el tiempo

En esta sección, se presentan diversos modos de modelizar la coherencia temporal usada en la literatura.

A lo largo del texto los símbolos en **negrita** se usarán para indicar cantidades de naturaleza vectorial así como matrices. Los símbolos sin negrita indicarán cantidades escalares. Cabe anotar que no se hará distinción

45 cuando se discutan ejemplos de videos 1D, y en estos casos se usarán símbolos sin negrita.

Considérese un dominio espacio- temporal continuo $\Omega \times [0, T]$ donde $\Omega \subset \mathbb{R}^2$ es un dominio rectangular, y $T > 0$, y el dominio de edición $O \subset \Omega \times [0, T]$ con frontera suave. En algunos lugares del texto, para impedir ecuaciones engorrosas, se usará Ω^T a modo de notación abreviada para el dominio de video $\Omega \times [0, T]$. Cada sección temporal de O será denotada por $O_t = \{x \in \Omega: (x, t) \in O\}$. De manera similar, las porciones temporales

50 de $\Omega \times [0, T]$ se denotan por $\Omega_t: t \in [0, T]$, representando los fotograma del video continuo. Una ilustración de estos dominios se puede ver en la Figura 1.

Sea $u: \Omega \times [0, T] \rightarrow \mathbb{R}$ un video de valores escalares dado y sea $v: \Omega \times [0, T - 1] \rightarrow \mathbb{R}^2$ el campo de movimiento correspondiente. El valor del campo de movimiento en $(x, t) \in \Omega \times [0, T - 1]$, $v(x, t)$ representa la velocidad de

la proyección de una partícula en la escena 3D dentro del plano 2D de la imagen [12]. La trayectoria de la partícula se puede obtener resolviendo la siguiente ecuación diferencial ordinaria (ODE):

$$\frac{dx}{dt}(t) = v(x(t), t) \quad (2)$$

5 Donde $t \in [0, T]$. Por simplicidad se asume en este capítulo que las funciones se pueden diferenciar tantas veces como sea necesario.

Ejemplo 1: Suposición de constancia de brillo

Para un objeto *Lambertiano* bajo iluminación constante uniforme, el brillo de una partícula del objeto no cambia con el tiempo, lo cual implica que $u(x, t)$ sea constante a lo largo de las trayectorias. Esto se puede expresar en los términos de la derivada total a lo largo de una trayectoria dando como resultado la siguiente ecuación de constancia de brillo [13]:

$$0 = \frac{d}{dt} u(x(t), t) = \nabla u(x(t), t) \cdot v(x(t), t) + \frac{\partial u}{\partial t}(x(t), t) \quad (3)$$

Donde $\frac{d}{dt} u$ es la derivada de u a lo largo de las trayectorias $x(t)$ y ∇u se refiere al gradiente espacial de u . Defínase la *derivada convectiva* como:

$$\partial_v u(x, t) := \nabla u(x, t) \cdot v(x, t) + \frac{\partial u}{\partial t}(x, t) \quad (4)$$

15 La derivada convectiva expresa la derivada a lo largo de las trayectorias. Esta notación se usa para simplificar las expresiones y hacer explícita su conexión con el campo v . Se puede imponer la constancia del brillo mediante un funcional de energía tal como (1), en la cual el término de coherencia temporal penaliza la derivada convectiva:

$$E^{bc} = \int_0^T \int_{\Omega_t} \partial_v u(x, t)^2 dx dt \quad (5)$$

20 Al penalizar la derivada convectiva, esta energía impone que el vídeo varíe suavemente a lo largo de las trayectorias de movimiento.

Hay diversas opciones para discretizar esta energía [18]. Una forma es usar el flujo óptico *adelantado* a partir de t a $t + 1$, v^f . La derivada convectiva *adelantada* discreta se define entonces en un punto $(x, t) \in \Omega \times \{0, 1, \dots, T\}$ como sigue

$$25 \quad \partial_v^f u(x, t) := u(x + v^f(x, t), t + 1) - u(x, t) \quad (6)$$

Cuando el flujo óptico tiene una resolución de subpíxel, entonces $u(x + v^f(x, t), t + 1)$ tiene que ser interpolado, por ejemplo usando interpolación bilineal. Con esta definición, la energía anterior es:

$$E^{bc} = \sum_{t=0}^T E_{t,t+1}^{bc}(u_t, u_{t+1}) = \sum_{t=0}^{T-1} \sum_{x \in \Omega} \left(\partial_v^f u(x, t) \right)^2 \quad (7)$$

30 La suposición de constancia del brillo se ha usado extensamente para el cálculo del flujo óptico [24,3], y recientemente para interpolación de video y edición dado un flujo óptico calculado. En [15] y [20] se usa para llevar a cabo la eliminación de un objeto, por ejemplo segmentado por un usuario. Con el fin de eliminarlo, se tiene que recuperar el fondo ocluido. Esto se lleva a cabo propagando el fondo visible a lo largo de las trayectorias de movimiento. La derivada convectiva es discretizada usando el flujo óptico adelantado.

35 Para la edición de video con coherencia temporal del gradiente, los autores en [5] usan un gradiente 3D donde la componente temporal viene dada mediante la derivada convectiva. El trabajo aborda las diversas tareas de edición de video. En el trabajo realizan dos etapas: primero se propaga la información del color (usando la Estructura a partir del Movimiento). El resultado es consistente en el tiempo en el dominio de edición, pero pueden apreciarse uniones artificiales en el dominio espacial. Para remediar esto, se lleva a cabo una segunda etapa. Usando el gradiente espacial de la información propagada, se propone un funcional de energía con dos términos. Un término que lleva a cabo la edición mediante la ecuación de Poisson en cada fotograma imponiendo consistencia espacial; y, un término que filtra a lo largo de las trayectorias de movimiento para asegurar adicionalmente la consistencia en el tiempo. Estos dos términos son ponderados por un parámetro positivo. El video resultante es coherente espacialmente y temporalmente. Este trabajo se ha elaborado adicionalmente dentro de un marco de referencia completo en [4] para filtrado de imagen y video.

Un procedimiento similar de dos etapas se aplica en [10]. La diferencia con [5] es que la primera etapa consistente en la propagación de colores se hace usando la derivada convectiva (en lugar de usar la estructura a partir del movimiento). En [10] la derivada convectiva es discretizada alternando entre los flujos ópticos adelantados y retrasados, lo que hace posible la propagación a través un gran número de fotogramas sin degradación. Después, se usa una energía de dos términos similar a la de [4], para eliminar las uniones artificiales en el dominio espacial de una forma coherente en el tiempo.

Una aproximación relacionada interesante se sigue en [7], donde los autores integran explícitamente el flujo óptico, calculando un conjunto de trayectorias que cubren el dominio a editar. Estas trayectorias se usan entonces para propagar la información del color conocida dentro del dominio de edición. Para el cálculo de estas trayectorias, el flujo óptico en sí mismo tiene que interpolarse en las posiciones subpíxel. Ya que el flujo óptico es en general más regular que los fotogramas, los errores de interpolación que aparecen son más pequeños y menos perceptibles en el video resultante, pero pueden aún acumularse ocasionando una desviación considerable. Este problema se analiza en [9] donde se propone una solución elegante.

Finalmente, hay que mencionar la aproximación de *mosaicos desenvueltos* de [17], la cual es interesante debido a que evita el cálculo de movimiento fotograma a fotograma. En lugar de ello, los autores proponen calcular una textura estática desenvuelta, una secuencia de máscaras de oclusión, y una secuencia de transformaciones a partir de la textura desenvuelta en cada uno de los fotogramas en el video. Luego se lleva a cabo la edición directamente en la textura desenvuelta, y los cambios se mapean de regreso a la secuencia de video usando las transformaciones calculadas. La técnica de los mosaicos desenvueltos permite manejar un amplio rango de situaciones que incluyen enfoques, deformaciones geométricas y oclusiones. El método se basa sin embargo en una maquinaria algorítmica sustancial que incluye segmentación de video exacta, seguimiento de puntos clave y optimización no lineal para calcular la textura y el mapeado. También, dado que los mosaicos están fijados, los cambios de iluminación deben manejarse en una etapa de post procesado.

Ejemplo 2: Suposición del cambio de brillo global (GBC):

Bajo cambios de iluminación, no se mantiene la suposición de constancia del brillo. En esta sección esta suposición se generaliza para tener en cuenta cambios de iluminación aditivos espacialmente constantes. En ese caso, si se siguen las trayectorias de dos partículas, la diferencia de sus colores permanece constante. La suposición del cambio de brillo global se basa en esta observación básica. Se consideran dos partículas que en el tiempo t están en posiciones $x_0 \in \Omega$ y $y_0 \in \Omega$. Sus trayectorias son indicadas por $\varphi(x_0, s)$ y $\varphi(y_0, s)$ con $s \in [0, T]$. Luego para $k > 0$,

$$u(\varphi(y_0, t + k), t + k) - u(\varphi(x_0, t + k), t + k) = u(y_0, t) - u(x_0, t) \quad (8)$$

Esto se representa mediante la Figura 3. Después de reorganizar los términos, dividiendo por k y tomando $k \rightarrow 0$ se obtiene en consecuencia $\partial_v u(y_0, t) = \partial_v u(x_0, t)$. Dado que esto se mantiene para todas las $x_0, y_0 \in \Omega$, se obtiene que $\partial_v u(x, t)$ es una función de t , que es:

$$\partial_v u(x, t) = g(t) \quad (9)$$

Aquí $g(t)$ expresa la tasa de cambio de iluminación global. En consecuencia, (9) generaliza el patrón de constancia de brillo tomando en consideración cambios globales en la iluminación.

Tomando el gradiente espacial en ambos lados de la Ecuación (9), se encuentra una versión diferencial de (8):

$$\nabla \partial_v u(x, t) = \mathbf{0} \quad (10)$$

Los autores en [18] proponen un método variacional para la edición de video con base en esta suposición. Proponen una energía para propagación a lo largo de las trayectorias de flujo óptico penalizando desviaciones de la condición (10):

$$E^{gbc}(u) = \int_0^T \int_{\Omega_t} \|\nabla \partial_v u(x, t)\|^2 dx dt \quad (11)$$

Mientras que la Ecuación (10) implica un cambio de iluminación espacial constante, el modelo variacional permite cierta variación espacial en $\partial_v u$. Esta es una característica útil en aplicaciones prácticas ya que representa fuentes de luz, sombras y reflejos localizados, a la vez que se manifiesta en la frontera del dominio de edición.

Esta energía puede ser discretizada de muchas formas, dependiendo de las discretizaciones usadas para el gradiente y los operadores de derivada convectiva [18]. Por ejemplo, se puede usar el flujo óptico adelantado

para discretizar la derivada convectiva como antes, y usar una discretización estándar del gradiente mediante las diferencias finitas adelantadas de la derivada convectiva.

$$\nabla^f u(x, t) = [u(x + e_0, t) - u(x, t), u(x + e_1, t) - u(x, t)]^T \quad (12)$$

Donde $e_0 = (1,0)^T$ y $e_1 = (0,1)^T$. Con esta definición, la energía se puede escribir como una suma de los términos que acoplan dos fotogramas consecutivos $E_{t,t+1}^{gbc}(u_t, u_{t+1})$.

Esta energía se usó en [18] para una tarea de edición de video consistente en la modificación de la superficie de los objetos en el video (por ejemplo para agregar, modificar o eliminar un logotipo). Un usuario lleva a cabo la edición en al menos un fotograma clave, y este se propaga entonces al resto de los fotogramas mediante la minimización de E^{gbc} . La propagación obtenida se adapta a los cambios de iluminación los cuales son espacialmente suaves.

Ejemplo 3: La suposición de constancia de gradiente

Aun otra alternativa es asumir que el gradiente es constante a lo largo de trayectorias de movimiento, cual se puede expresar como:

$$\partial_v \nabla u(x, t) = \mathbf{0} \quad (13)$$

En la literatura se hace referencia a esto como la suposición de constancia de gradiente [23, 24, 16]. De forma similar a las ecuaciones (5) y (15), se puede definir un término de energía de coherencia temporal penalizando las desviaciones de la Ecuación (13):

$$E^{gc}(u) = \int_0^T \int_{\Omega_t} \|\partial_v \nabla u(x, t)\|^2 dx dt \quad (14)$$

Con el gradiente discreto y la derivada convectiva usados anteriormente, se obtiene una versión discreta de la energía que se puede expresar como un sumatorio de los términos que acoplan dos fotogramas sucesivos, $E_{t,t+1}^{gc}(u_t, u_{t+1})$.

Esta suposición es más restrictiva que la suposición del cambio de brillo global, ya que solo se mantiene cuando el movimiento es una traslación global. Por esta razón, este patrón no se ha usado hasta ahora para la edición de video, y se ha usado principalmente para el cálculo del flujo óptico [23].

Referencias:

[1] R.A. Adams. Sobolev spaces. Academic Press, New York, 1975.

[2] A. Ayvaci, M. Raptis, and S. Soatto. Sparse occlusion detection with optical flow. International Journal of Computer Vision (IJCV), 97(3):322–338, May 2012.

[3] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A Database and Evaluation Methodology for Optical Flow. International Journal of Computer Vision, 92(1):1–31, 2011. 20

[4] P. Bhat, C. L. Zitnick, M. Cohen, and B. Curless. Gradientshop: A gradient-domain optimization framework for image and video filtering. ACM Transactions on Graphics, 29:1–14, April 2010.

[5] P. Bhat, C.L. Zitnick, N. Snavely, A. Agarwala, M. Agrawala, M.F. Cohen, B. Curless, and S.B. Kang. Using photographs to enhance videos of a static scene. In Proceedings of the Eurographics Symposium on Rendering Techniques, pages 327–338. Eurographics Association, 2007.

[6] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 33(3):500–513, 2011.

[7] A. Bugeau, P. Gargallo, O. D'Hondt, A. Hervieu, N. Papadakis, and V. Caselles. Coherent Background Video Inpainting through Kalman Smoothing along Trajectories. In Modeling, and Visualization Workshop, page 8,

2010.

[8] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.

[9] T. Crivelli, M. Fradet, P.-H. Conze, P. Robert, and P. Perez. Robust optical flow integration. *Image Processing, IEEE Transactions on*, 24(1):484–498, Jan 2015.

[10] G. Facciolo, R. Sadek, A. Bugeau, and V. Caselles. Temporally consistent gradient domain video editing. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, pages 59–73. Springer-Verlag, 2011.

[11] The Foundry. Nuke. <https://www.thefoundry.co.uk/products/nuke/>

10 [12] B.K.P. Horn. *Robot Vision*. Electrical Engineering and Computer Science. MIT Press, 1986.

[13] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[14] J. Jia, Y.-W. Tai, T.-P. Wu, and C.-K. Tang. Video repairing under variable illumination using cyclic motions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(5):832–9, 2006.

15 [15] A.C. Kokaram, B. Collis, and S. Robinson. Automated rig removal with bayesian motion interpolation. *IEEE Journal on Vision, Image and Signal Processing*, 152:407–414, Aug 2005.

[16] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *International Journal of Computer Vision (IJCV)*, 67(2):141–158, April 2006.

[17] A. Rav-Acha, P. Kohli, C. Rother, and A. Fitzgibbon. Unwrap mosaics: A new representation for video editing. *ACM Transactions on Graphics (SIGGRAPH 2008)*, August 2008, 21

20 [18] R. Sadek, G. Facciolo, P. Arias, and V. Caselles. A variational model for gradient-based video editing. *International Journal of Computer Vision*, 103(1):127–162, 2013.

[19] Mike Seymour. The art of optical flow. https://www.fxguide.com/featured/art_of_optical_flow/, February 2006.

25 [20] T. Shiratori, Y. Matsushita, X. Tang, and S.B. Kang. Video completion by motion field transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 411–418, 2006.

[21] D. Sun, S. Roth, and M.J. Black. Secrets of optical flow estimation and their principles. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2439, 2010.

[22] Imagineer Systems. Mocha. <http://imagineersystems.com/>.

30 [23] S. Uras, F. Girosi, A. Verri, and V. Torre. A computational approach to motion perception. *Biological Cybernetics*, 60:79–87, 1988.

[24] J. Weickert, A. Bruhn, T. Brox, and N. Papenberg. A survey on variational optic flow methods for small

displacements. In O. Scherzer, editor, *Mathematical Models for Registration and Applications to Medical Imaging*, volume 10 of *Mathematics in Industry*. Springer, Berlin, 2006.

[25] Y. Zhang, J. Xiao, and M. Shah. Motion Layer Based Object Removal in Videos. In *7th IEEE Workshops on Application of Computer Vision*, 2005.

5 Breve descripción de la invención

De acuerdo con las limitaciones de la técnica anterior, sería deseable reducir la cantidad de tiempo y cálculos necesarios para editar un video de una manera viable.

10 La presente invención describe un método implementado por ordenador y un sistema para la minimización eficiente de energías que aparecen comúnmente en las aplicaciones de edición de video, construyendo una serie de energías que afectan a un solo fotograma que se pueden minimizar independientemente, y cuyos mínimos se aproximan a los mínimos de la energía original.

15 En particular, la presente invención es útil en la edición de video que usa modelos de movimiento no paramétricos tales como el flujo óptico. Sin embargo, también es válida para modelos de movimiento paramétricos, ya que un flujo óptico puede estar directamente asociado a cualquier transformación entre dos imágenes.

En consecuencia, es un objeto de la presente invención, un método y un sistema implementados por ordenador que se pueden aplicar a la edición que afectan a un video o la edición de una secuencia de video con coherencia temporal.

El método implementado por ordenador incluye al menos las etapas de:

- 20 i) calcular un campo de movimiento que modele la coherencia temporal entre los fotogramas sucesivos;
- ii) definir un funcional de energía que modele las propiedades deseadas para ser impuestas en la secuencia de video resultado;
- iii) dividir la secuencia de video en dos conjuntos, un conjunto con fotogramas pares y otro conjunto con fotogramas impares;
- 25 iv) calcular el campo de movimiento entre los fotogramas consecutivos en las secuencias divididas;
- v) de manera recursiva llevar a cabo las etapas iii) y iv) hasta que los conjuntos que se van a dividir contengan un único fotograma para editar;
- vi) minimizar el funcional de energía (o si es necesario una versión adaptada del funcional de energía) para cada conjunto que contenga un fotograma para editar;
- 30 vii) integrar los fotogramas editados y extraer la secuencia de video editada

Opcionalmente, el método implementado por ordenador puede incluir una etapa previa de identificación de un fotograma en una secuencia de video la cual tiene un objeto editado para ser propagado.

Opcionalmente, el funcional de energía incluye un sumatorio de energías de edición dependientes de los fotogramas de manera independiente y de las energías que acoplan temporalmente fotogramas consecutivos.

35 Opcionalmente, el funcional de energía puede estar basado en una suposición de cambio de brillo global, una suposición de constancia de brillo o una suposición de constancia de gradiente.

40 En cuanto al sistema, es adecuado para editar una secuencia de video con coherencia temporal e incluye un primer medio de procesamiento para calcular un campo de movimiento que aporte coherencia temporal entre fotogramas sucesivos y para definir un funcional de energía que modele las propiedades para ser impuestas en la secuencia de video, un segundo medio de procesamiento para dividir la secuencia de video en dos conjuntos con fotogramas pares y fotogramas impares, en donde se lleva a cabo la división recursivamente, hasta que haya un único fotograma a editar en cada conjunto, un tercer medio de procesamiento para calcular el campo de movimiento entre dos fotogramas consecutivos en cada conjunto después de cada división, un cuarto medio de procesamiento para minimizar el funcional de energía en cada conjunto para obtener los

45 fotogramas editados, un quinto medio de procesamiento para integrar los fotogramas editados y extraer toda

la secuencia de video editada. Cualquier medio de procesamiento debe interpretarse en la presente invención como cualquier dispositivo electrónico que comprende un procesador.

Opcionalmente, el sistema puede incluir un sexto medio de procesamiento para identificar un fotograma en una secuencia de video que tiene un objeto editado para propagar.

5 Los medios de procesamiento anteriores deberían interpretarse como cualquier dispositivo o dispositivos de cálculo multipropósito para procesamiento y administración de datos. En particular, estos medios de procesamiento pueden implementarse como uno o más dispositivos de cálculo electrónicos que incluyen, sin limitación, un ordenador de mesa, un ordenador portátil, un servidor de red y similares.

10 Las realizaciones preferidas de la presente invención serán descritas a manera solo de ejemplo con referencia en los dibujos acompañantes.

Breve descripción de los dibujos

Se describen brevemente a continuación una serie de dibujos los cuales ayudan a un mejor entendimiento de la invención y están relacionados expresamente con una realización de dicha invención, presentada como un ejemplo no limitante de la misma.

15 Fig. 1: Ilustración del dominio de edición dentro del dominio de video. También muestra la porción temporal en el momento t .

Fig. 2: Diagrama del procedimiento de División y Minimización.

Fig. 3: Diagrama del procedimiento de División de Video Impar.

Fig. 4: Diagrama del procedimiento de División de Video Par.

20 Fig. 5: Diagrama del procedimiento de División del Movimiento Impar.

Fig. 6 Diagrama del procedimiento de División del Movimiento Par.

Fig. 7: Diagrama del procedimiento de Integración.

25 Fig. 8: Ejemplo del tipo de aplicación de edición de video que se puede acelerar mediante el método divulgado. La primera fila es una secuencia f original para editar, la segunda fila muestra dicha secuencia f con el primer y último fotograma editados manualmente. La tercera fila es la solución deseada calculada automáticamente.

Fig. 9: Diagrama de bloques de los elementos principales de la invención.

Descripción detallada de la invención

30 La presente realización describe un método para reducir la complejidad y resolver el tiempo de los procesos de edición de video asumiendo coherencia temporal. Sea $\Omega \times [1, T]$ el dominio (del video espacio temporal) donde $\Omega \subset \mathbb{R}^2$ es el dominio espacial, y $T \geq 1$ representa el número de fotogramas que componen el video. Sea $u: \Omega \times [1, T] \rightarrow \mathbb{R}^M$ un video a color ($M > 1$) o en nivel de gris ($M = 1$) tal que $u(x, t)$ representa el valor del píxel en la ubicación x del fotograma t . Además, sea $v: \Omega \times [1, T - 1] \rightarrow \mathbb{R}^2$ el correspondiente campo de movimiento. Este campo de movimiento da una correspondencia entre píxeles en el fotograma t y $t + 1$ de tal forma que $u(x, t) = u(x + v(x), t + 1)$ (bajo la suposición de constancia de brillo), esto es, v establece la
35 coherencia temporal en el video.

A manera de ejemplo, a continuación se definen diversos algoritmos escritos en pseudocódigo, para un mejor entendimiento. Estos algoritmos se pueden implementar en cualquier dispositivo de procesamiento (por ejemplo un ordenador) para editar una secuencia de video de acuerdo con los principios de la invención. Especialmente, el núcleo de la invención se puede ver en la secuencia de las etapas que se toman en el
40 Algoritmo 2.

Algoritmo 1: $u \leftarrow \text{Minimizar}(J, f, v)$

Requiere:

Un video en escala de grises o a color $f(x, t), t \in [1, T]$ compuesto por T fotogramas. El campo de movimiento $v(x, t), t \in [1, T - 1]$

45 Un funcional de energía $J(u, v, f)$ para minimizar o la ecuación en derivadas parciales que involucra

el video dado f y el campo de movimiento v .

Proporciona: Un video editado $u(x, t), t \in [1, T]$.

1: $u \leftarrow$ mínimo de $J(u, v, f)$ con respecto a u

5 **Algoritmo 2:** $u \leftarrow$ Dividir y Minimizar (J, f, v)

Requiere:

Un video en escala de grises o a color $f(x, t), t \in [1, T]$ compuesto por T fotogramas. El campo de movimiento $v(x, t), t \in [1, T - 1]$

Un funcional de energía $J(u, v, f)$ para minimizar o la ecuación en derivadas parciales que involucra el video f dado y el campo de movimiento v .

10

Proporciona: Un video editado $u(x, t), t \in [1, T]$

1. nCuadros \leftarrow número de fotogramas
2. **si** nCuadros es 1 entonces
 - a. $u \leftarrow$ Minimizar (j, f, v) con respecto a u
3. **de otro modo**{nCuadros>1}
 - a. $f^{impar}(x, t) \leftarrow$ División de Video Impar (f)
 - b. $f^{par}(x, t) \leftarrow$ División de Video Par (f)
 - c. $v^{impar}(x, t) \leftarrow$ División de Movimiento Impar (v)
 - d. $v^{par}(x, t) \leftarrow$ División de Movimiento Par (f)
 - e. $u^{impar} \leftarrow$ Dividir y Minimizar (J, f^{impar}, v^{impar})
 - f. $u^{par} \leftarrow$ Dividir y Minimizar (J, f^{par}, v^{par})
 - g. $u \leftarrow$ Integrar(u^{impar}, u^{par})
4. **terminar si**
5. **retornar** u

15

20

25

Algoritmo 3: $f^{impar} \leftarrow$ División de Video Impar (f)

Requiere:

Un video en escala de grises o a color $f(x, t), t \in [1, T]$ compuesto por T fotogramas

Proporciona: Un video $f^{impar}(x, t)$ compuesto por los fotogramas impares a partir de f

30

1. nCuadros \leftarrow número de fotogramas
2. **si** nCuadros es 1 entonces
 - a. $f^{impar} \leftarrow f$
3. **de otro modo**{nCuadros>1}
 - a. $j \leftarrow 1$
 - b. **para** $i = 1$ a $i = nCuadros$ **hacer**
 - i. **si** i es impar entonces
 1. $f^{impar}(x, j) = f(x, i)$
 2. $j \leftarrow j + 1$
 - ii. **terminar si**
 - iii. $i \leftarrow i + 1$
 - c. **terminar para**

35

40

4. **terminar si**
5. **retornar** f^{impar}

Algoritmo 4: $f^{par} \leftarrow$ División de Video Par (f)

5 **Requiere:**

Un video en escala de grises o a color $f(x, t), t \in [1, T]$ compuesto por T fotogramas

Proporciona: Un video $f^{par}(x, t)$ compuesto por los fotogramas impares a partir de f

1. nCuadros \leftarrow número de fotogramas
2. **si** nCuadros es 1 **entonces**
 - 10 a. $f^{par} \leftarrow f$
3. **de otro modo** {nCuadros>1}
 - a. $j \leftarrow 1$
 - b. **para** $i = 1$ a $i = nCuadros$ **hacer**
 - i. **si** i es par **entonces**
 - 15 1. $f^{par}(x, j) = f(x, i)$
 2. $j \leftarrow j + 1$
 - ii. **terminar si**
 - iii. $i \leftarrow i + 1$
 - c. **terminar para**
4. **terminar si**
- 20 5. **retornar** f^{par}

Algoritmo 5: $v^{impar} \leftarrow$ División de Movimiento Impar (v)

Requiere:

- 25 Un campo de movimiento $v(x, t), t \in [1, T - 1]$ a partir de un video compuesto de T fotogramas.
El número de fotogramas nCuadros del correspondiente video.

Proporciona: Un campo de movimiento $v^{impar}(x, t)$ que debe ser coherente con la División de Video Impar (f)

1. nCuadros \leftarrow número de fotogramas
- 30 2. **si** nCuadros es 1 **entonces**
 - a. $v^{impar} \leftarrow v$
3. **de otro modo** {nCuadros>1}
 - a. $j \leftarrow 1$
 - b. **para** $i = 1$ a $i = nCuadros - 1$ **hacer**
 - i. **si** i es impar **entonces**
 - 35 1. $v^{impar}(x, j) \leftarrow v(x, i) + v(x + v(x, i), i + 1)$
 2. $j \leftarrow j + 1$
 - ii. **terminar si**
 - iii. $i \leftarrow i + 1$
 - c. **terminar para**
- 40

4. **terminar si**
5. **retornar** v^{impar}

Algoritmo 6: $v^{par} \leftarrow$ División de Movimiento Par (v)

5 **Requiere:**

Un campo de movimiento $v(x, t), t \in [1, T - 1]$ a partir de un video compuesto por T fotogramas.
El número de fotogramas nCuadros del video correspondiente.

Proporciona: Un campo de movimiento $v^{par}(x, t)$ que podría ser coherente con la División de Video Par (f)

1. nCuadros \leftarrow número de fotogramas
- 10 2. **si** nCuadros es 1 **entonces**
 - a. $v^{par} \leftarrow v$
3. **de otro modo** {nCuadros>1}
 - a. $j \leftarrow 1$
 - b. **para** $i = 1$ a $i = nCuadros - 1$ **hacer**
 - 15 i. **si** i es par **entonces**
 1. $v^{par}(x, j) \leftarrow v(x, i) + v(x + v(x, i), i + 1)$
 2. $j \leftarrow j + 1$
 - ii. **terminar si**
 - iii. $i \leftarrow i + 1$
 - 20 c. **terminar para**
4. **terminar si**
5. **retornar** v^{par}

Algoritmo 7: $u \leftarrow$ Integrar (u^{impar}, u^{par})

25 **Requiere:**

Dos secuencias de video en nivel de gris o a color (u^{impar}, u^{par}).

La diferencia en el número de fotogramas no puede ser mayor que uno.

Asegurar: Un nuevo video u compuesto por u^{impar} y u^{par}

1. nCuadros^{impar} \leftarrow número de fotogramas de u^{impar}
- 30 2. nCuadros^{par} \leftarrow número de fotogramas de u^{par}
3. $i \leftarrow 1$
4. $j \leftarrow 1$
5. **a la vez que** $i < nCuadros^{impar}$ o $i < nCuadros^{par}$ **hacer**
 - a. $u(x, j) \leftarrow u^{impar}(x, i)$
 - 35 b. $u(x, j + 1) \leftarrow u^{par}(x, i)$
 - c. $i \leftarrow i + 1$
 - d. $j \leftarrow j + 2$
6. **terminar a la vez que**

Ahora el siguiente problema de edición de video se considera como un ejemplo de cómo proceder para resolver el problema de acuerdo con la invención: sea f un video a color compuesto por 5 fotogramas, como se muestra en la primera fila de la Fig. 8, en la cual los fotogramas 1 y 5 se han editado manualmente cambiando dos lados de la caja (segunda fila). La idea es propagar automáticamente la información del primer y el último fotograma al resto de fotogramas 2, 3 y 4, como se muestra en la Fig. 8, tercera fila.

Uno de los métodos posibles en la literatura para resolver este problema se explica a continuación. Sea $O \subset \Omega \times [1,3]$ el dominio de edición (O son los lados de la caja de la Figura 8) con frontera Lipschitz [1] (para simplificar, se puede considerar que O tiene un límite suave). Sea $O_t = \{x \in \Omega: (x, t) \in O, t \in [1,3]\}$, es decir, O_t es el área de edición del fotograma t . Una ilustración de estos dominios se puede ver en la Figura 1. Además, sea v el mapa de correspondencia entre los fotogramas.

El problema se puede resolver minimizando un funcional de energía. En este ejemplo, se usa la suposición del cambio de brillo global:

$$J(u) = \int_0^T \int_{\Omega_t} \|\nabla \partial_v u(x, t)\|^2 dx dt \quad (15)$$

Donde ∇ es el operador gradiente y ∂_v es la derivada convectiva. Siguiendo el cálculo variacional, el mínimo de energía (15) es la solución a la ecuación de Euler-Lagrange dada por la siguiente EDP (Ecuación en Derivadas Parciales) de cuarto orden

$$\partial_v^* \text{div} \nabla \partial_v u(x, t) = 0, \quad (x, t) \in O \quad (16)$$

donde div es la divergencia espacial adyacente a $-\nabla$ y ∂_v^* indica el operador adjunto de la derivada convectiva, dada por $\partial_v^* f = -\frac{\partial f}{\partial t} - \text{div}(vf)$.

Esta ecuación se completa con las condiciones de contorno Dirichlet,

$$u(x, t) = u_0(x, t), \quad x \in \Omega^t / O_t \quad (17)$$

De acuerdo con la propuesta actual, no se usa todo el video u_0 ni todo el mapa v de correspondencia. De una forma informal, el método aplicado se expresa como sigue a continuación: Las entradas al problema son el video f y la información de conectividad entre los fotogramas consecutivos. Esta información de conectividad (o campo de movimiento) es aproximada usualmente por el flujo v óptico.

La primera etapa se relaciona con la división de la secuencia de video de entrada para ser editada.

De acuerdo con esta etapa, el video de entrada se puede dividir en dos conjuntos. Los fotogramas impares y los fotogramas pares (Figs. 3 y 4 y los algoritmos 3 y 4). El campo de movimiento también tiene que ser dividido (Figs. 5 y 6 y los algoritmos 5 y 6). Esta etapa se tiene que llevar a cabo recursivamente hasta que las secuencias de video sólo tengan un solo fotograma. En el ejemplo se tienen 3 secuencias de video al final de la etapa recursiva de división, realmente no están compuestas de un único fotograma sino que afectan a un fotograma más las tapas (los fotogramas ya editados). Con respecto a esto, las secuencias se denominan de un fotograma debido a que hay solo un fotograma que contiene valores desconocidos. Una vez que se crean todos los videos compuestos de un fotograma más las tapas y sus correspondientes campos de movimiento, se resuelve el problema independientemente para cada video pequeño. Una vez que se resuelve el problema para cada video pequeño se hace necesaria una etapa de integración para componer la solución completa a partir de cada una de las soluciones individuales de los diversos problemas pequeños, esto se lleva a cabo siguiendo la Fig. 7 y el algoritmo 7. Esta integración puede también incluir algunas etapas de corrección para corregir pequeños posibles errores generados por los algoritmos 5 y 6.

Ahora, se describe paso a paso el algoritmo 2 en el contexto de ejemplo del problema de edición de video anterior para una secuencia de video $f=\{f_0, f_1, f_3, f_4\}$, con los fotogramas editados manualmente $\{f_0, f_4\}$ y el campo de movimiento $v=\{v_0, v_1, v_2, v_3\}$

1. $u \leftarrow$ Dividir y Minimizar (J, f, v) (debido que f tiene más de un fotograma para editar)

1.1. Dividir la secuencia de video original $f=\{f_0, f_1, f_2, f_3, f_4\}$ siguiendo los algoritmos 3 y 4.

$f_0 \leftarrow$ División de video Impar(f)

$$f_0 = \{f_0, f_1, f_3, f_4\}$$

$f_e \leftarrow$ División de video Par(f)

$$f_e = \{f_0, f_2, f_4\}$$

1.2. Calcular los nuevos campos de movimiento a partir de v siguiendo los algoritmos 5 y 6

$v_o \leftarrow$ División del Movimiento Impar (v)

5

$$v_o = \{v(x,0), v(x,1)+v(x+v(x,1),2), v(x,3)\}$$

$v_{oe} \leftarrow$ División del Movimiento Par (v)

$$v_e = \{v(x,0)+v(x+v(x,0),1), v(x,2)+v(x+v(x,2),3)\}$$

1.3. Resolver el problema para cada f_o , f_e y sus correspondientes campos de movimiento v_o , v_e .

$u_e \leftarrow$ Minimizar (J, f_e, v_e) (debido a que f_e tiene solo un solo fotograma para editar)

10

$$u_e = \{f_0, u_{e1}, f_4\}$$

$u_o \leftarrow$ Dividir y Minimizar (J, f_o, v_o) (debido a que f_o tiene más de un fotograma para editar)

1.3.1. dividir la secuencia de video f_o siguiendo los algoritmos 3 y 4

15

$f_{oo} \leftarrow$ División de Video Impar (f_o)

$$f_{oo} = \{f_0, f_1, f_4\}$$

$f_{oe} \leftarrow$ División de Video Par (f_o)

$$f_{oe} = \{f_0, f_3, f_4\}$$

1.3.2. Calcular los nuevos campos de movimiento a partir de v_o siguiendo los algoritmos 5 y 6

20

$v_{oo} \leftarrow$ División del Movimiento Impar(v_o)

$$v_{oo} = \{v_o(x,0), v_o(x,1)+v_o(x+v_o(x,1),2)\}$$

$v_{oe} \leftarrow$ División del Movimiento Par (v_o)

$$v_{oe} = \{v_o(x,0)+v_o(x+v(x,0),1), v_o(x,2)\}$$

1.3.3. Debido a que el número de fotogramas para editar de f_{oo} y f_{oe} es uno, se resuelve

25

$u_{oo} \leftarrow$ Minimizar (J, f_{oo}, v_{oo})

$$u_{oo} = \{f_0, u_{oo1}, f_4\}$$

$u_{oe} \leftarrow$ Minimizar (J, f_{oe}, v_{oe})

$$u_{oe} = \{f_0, u_{oe1}, f_4\}$$

1.3.4. Integrar las soluciones u_{oo} , u_{oe}

30

$u_o \leftarrow$ Integrar (u_{oo} , u_{oe})

$$u_o = \{f_0, u_{oo1}, u_{oe1}, f_4\}$$

1.3.5. Integrar las soluciones u_o , u_e

$u \leftarrow \text{Integrar}(u_o, u_e)$

$u = \{f_0, u_{o1}, u_{e1}, u_{oe1}, f_4\}$

1.3.6. Retornar el video editado u .

5 Como es evidente, los algoritmos anteriores se pueden codificar como instrucciones en un lenguaje de ordenador adecuado para llevar a cabo automáticamente las operaciones descritas cuando se ejecutan en un ordenador.

10 La Fig. 9 muestra un diagrama de bloques que representa los principales elementos funcionales que administran los datos en una realización. Las cajas **10-50** se pueden referir a unidades lógicas definidas en un ordenador u ordenadores en una red. Una secuencia (f) de video para editar se recibe por un primer medio **10** de procesamiento el cual calcula un campo (v) de movimiento que modela la consistencia en el tiempo entre los fotogramas (f_i, f_{i+1}) sucesivos y define además un funcional de energía (J) que modela las propiedades a cumplir por la secuencia (f) de video. Un segundo medio **20** de procesamiento divide recursivamente la secuencia (f) de video en dos conjuntos con fotogramas pares (f^{par}) y fotogramas impares (f^{impar}). Esto se lleva a cabo hasta que solo haya un fotograma para editar en cada conjunto. Un tercer medio **30** de procesamiento está a cargo de calcular el campo (v) de movimiento entre los fotogramas consecutivos en cada conjunto después de cada división. Un cuarto medio **40** de procesamiento minimiza el funcional de energía (J) en cada conjunto para obtener los fotogramas editados. Por último un quinto medio **50** de procesamiento integra los fotogramas editados y proporciona toda la secuencia (u) de video editada.

20 Aunque la invención se ha explicado en relación con su(s) realización(es) preferida(s) como se menciona anteriormente, se puede entender que se pueden hacer otras muchas modificaciones y variaciones sin apartarse del alcance de la presente invención definido mediante las reivindicaciones anexas.

REIVINDICACIONES

1. Un método implementado por ordenador para editar una secuencia de video con consistencia en el tiempo que comprende las etapas de:
 - 5 i) calcular un campo de movimiento que modele la consistencia temporal entre fotogramas sucesivos;
 - ii) definir un funcional de energía que modele las propiedades deseadas que debe poseer la secuencia de video;
 - 10 iii) dividir la secuencia de video en dos conjuntos, uno con fotogramas pares y otro con fotogramas impares; caracterizado por que el método comprende además las etapas:
 - 15 iv) calcular el campo de movimiento entre los fotogramas consecutivos en las secuencias divididas;
 - v) llevar a cabo recursivamente las etapas iii) y iv) hasta que los conjuntos que se dividan contengan un único fotograma para editar;
 - 20 vi) minimizar el funcional de energía para cada conjunto que contenga un fotograma para editar;
 - vii) integrar los fotogramas editados y extraer la secuencia completa de video editada.
 2. El método implementado por ordenador de la reivindicación 1, en donde este comprende una etapa previa de identificar un fotograma en una secuencia de video que tiene un objeto editado para propagar.
 - 25 3. El método implementado por ordenador de la reivindicación 1 o 2, en donde el funcional de energía comprende un sumatorio de energías para la edición que dependen de los fotogramas y las energías de acople temporal que dependen de los pares de fotogramas consecutivos.
 - 30 4. El método implementado por ordenador de cualquiera de las reivindicaciones 1 a la 3, en donde el funcional de energía se basa en al menos uno de los siguientes modelos:
 - suposición de cambio de brillo global;
 - suposición de constancia de brillo;
 - 35 - suposición de constancia de gradiente.
 5. Un sistema para editar una secuencia de video con consistencia en el tiempo que comprende:
 - 40 - un primer medio (10) de procesamiento para calcular un campo (v) de movimiento que modele consistencia en el tiempo entre los fotogramas (f_i, f_{i+1}) sucesivos y para definir un funcional (J) de energía que modele las propiedades que debe poseer la secuencia (f) de video;
 - 45 - un segundo medio (20) de procesamiento para dividir la secuencia (f) de video en dos conjuntos con fotogramas pares (f^{par}) y fotogramas impares (f^{impar}), en donde la división se hace de manera recursiva hasta que solo haya un fotograma para editar en cada conjunto;
 - un tercer medio (30) de procesamiento para calcular el campo (v) de movimiento entre los fotogramas consecutivos en cada conjunto después de cada división;
 - 50 - un cuarto medio (40) de procesamiento para minimizar el funcional (J) de energía en cada conjunto para obtener los fotogramas editados;
 - un quinto medio (50) de procesamiento para integrar los fotogramas editados y proporcionar la totalidad de la secuencia (u) de video editada.
 - 55
 6. Un sistema para editar una secuencia de video con consistencia temporal según la reivindicación 5, que comprende un sexto medio de procesamiento para identificar un fotograma en una secuencia de video que tiene un objeto editado para ser propagado.

7. Un producto de programa por ordenador para editar una secuencia de video con consistencia temporal comprendiendo instrucciones de código por ordenador almacenadas en el mismo que, cuando son ejecutadas por un procesador, ocasionan que el procesador lleve a cabo el método de cualquiera de las reivindicaciones 1 a la 4.

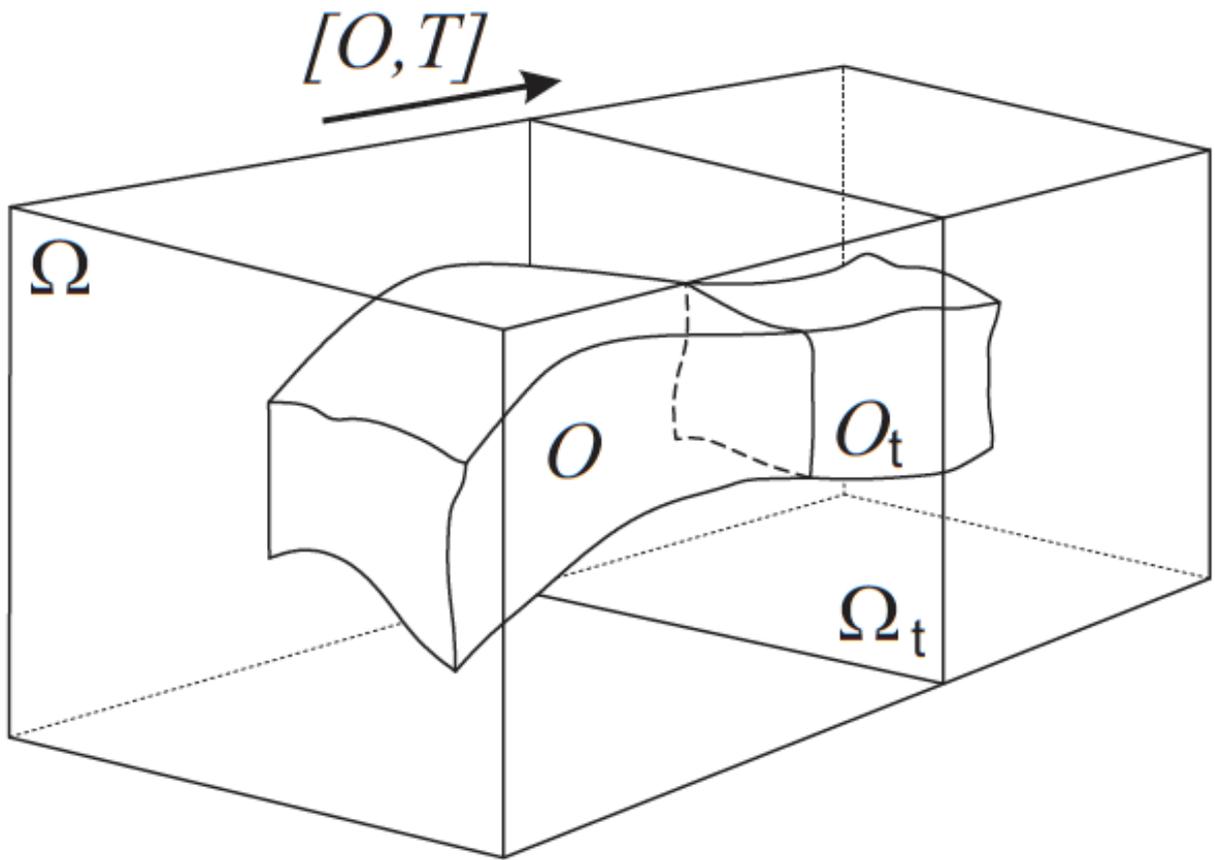


Fig. 1

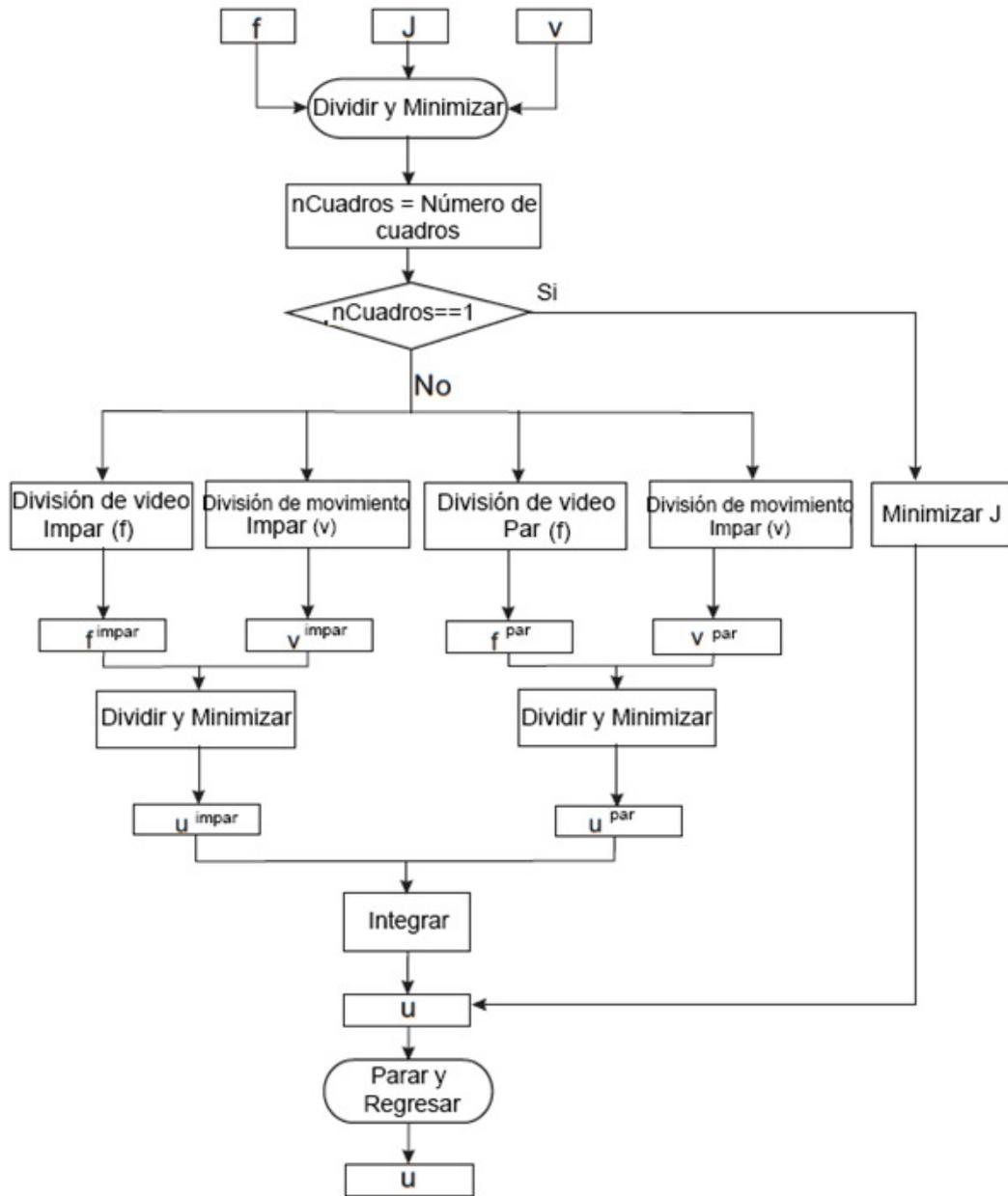


Fig. 2

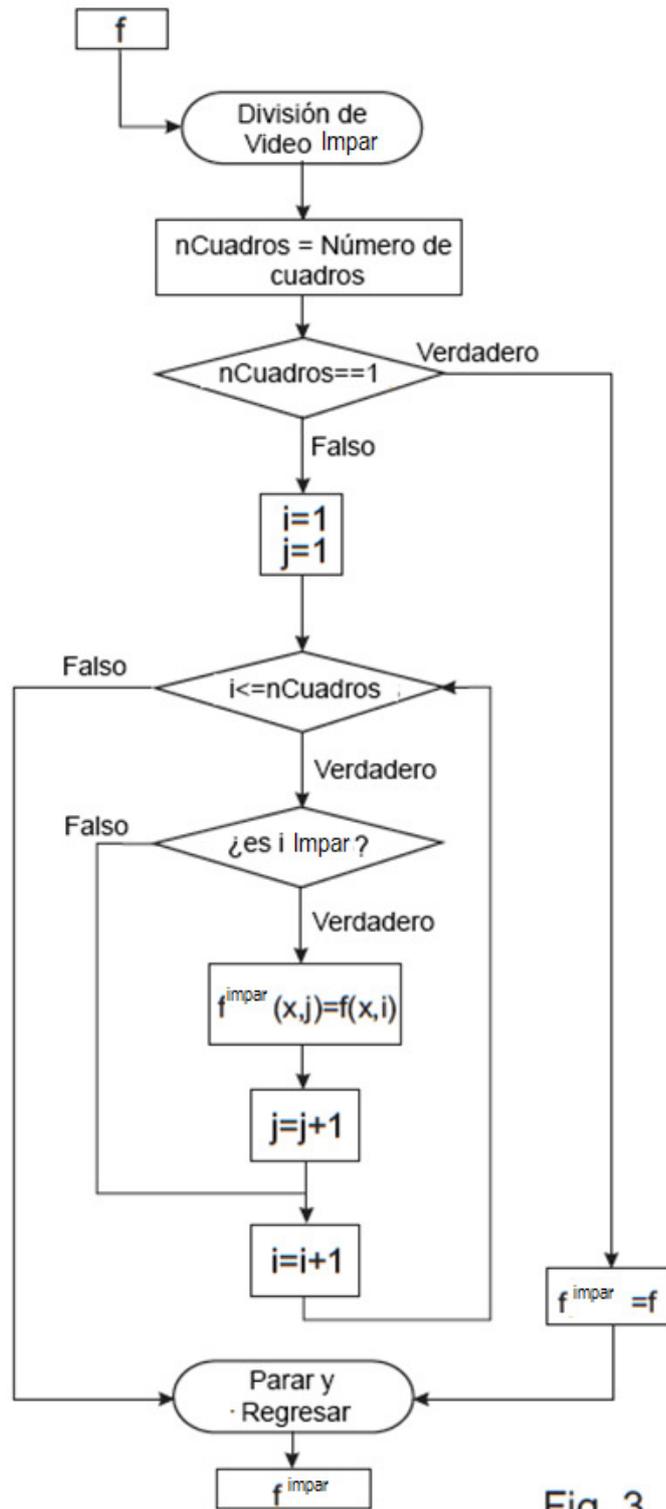


Fig. 3

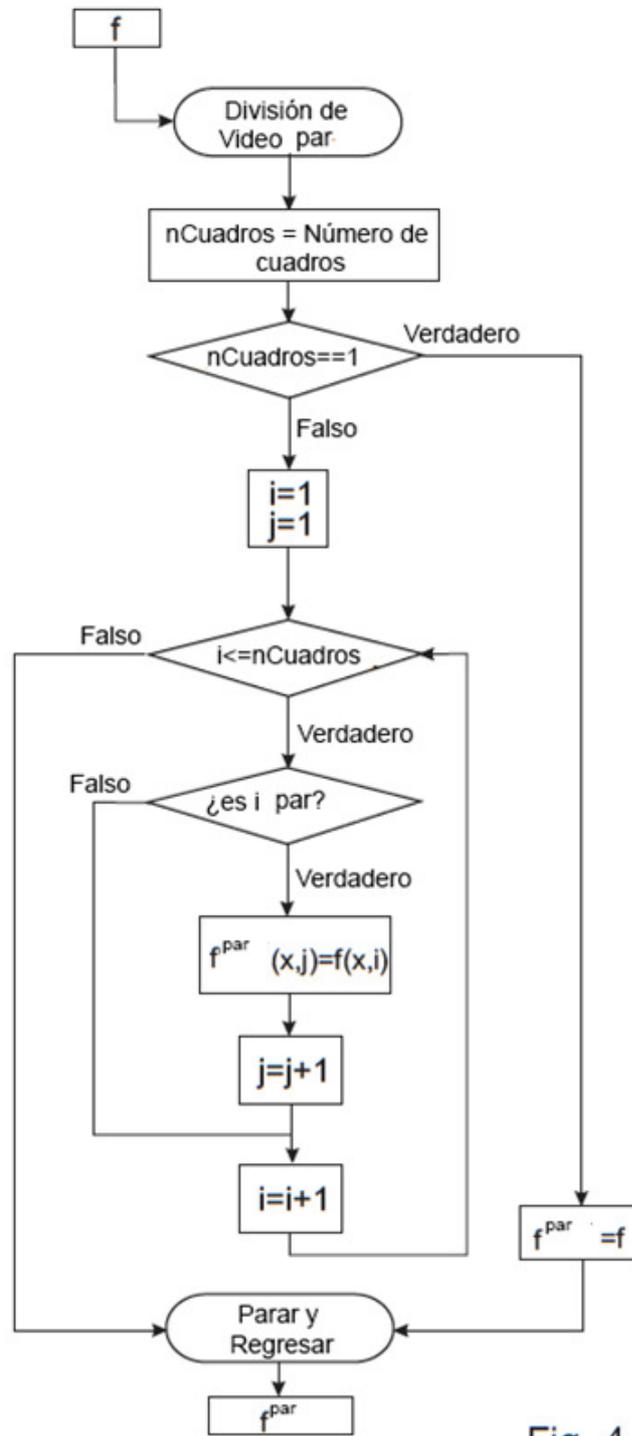


Fig. 4

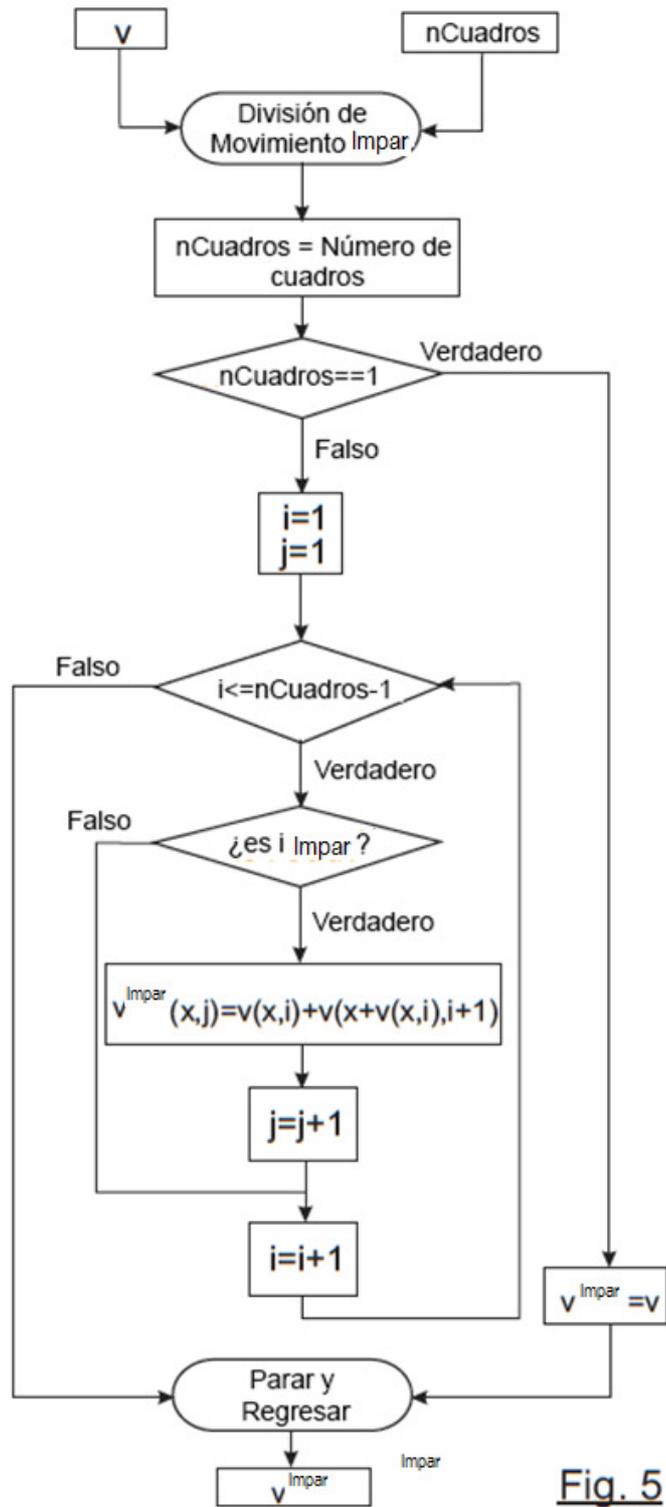


Fig. 5

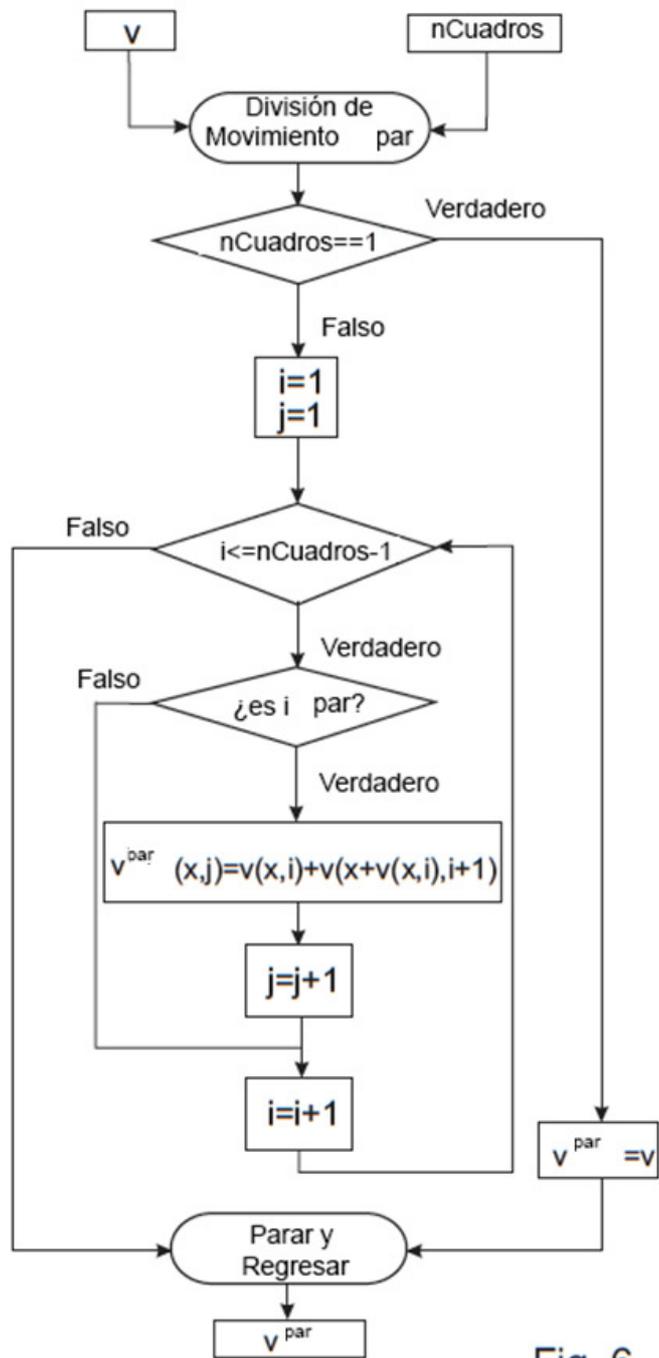


Fig. 6

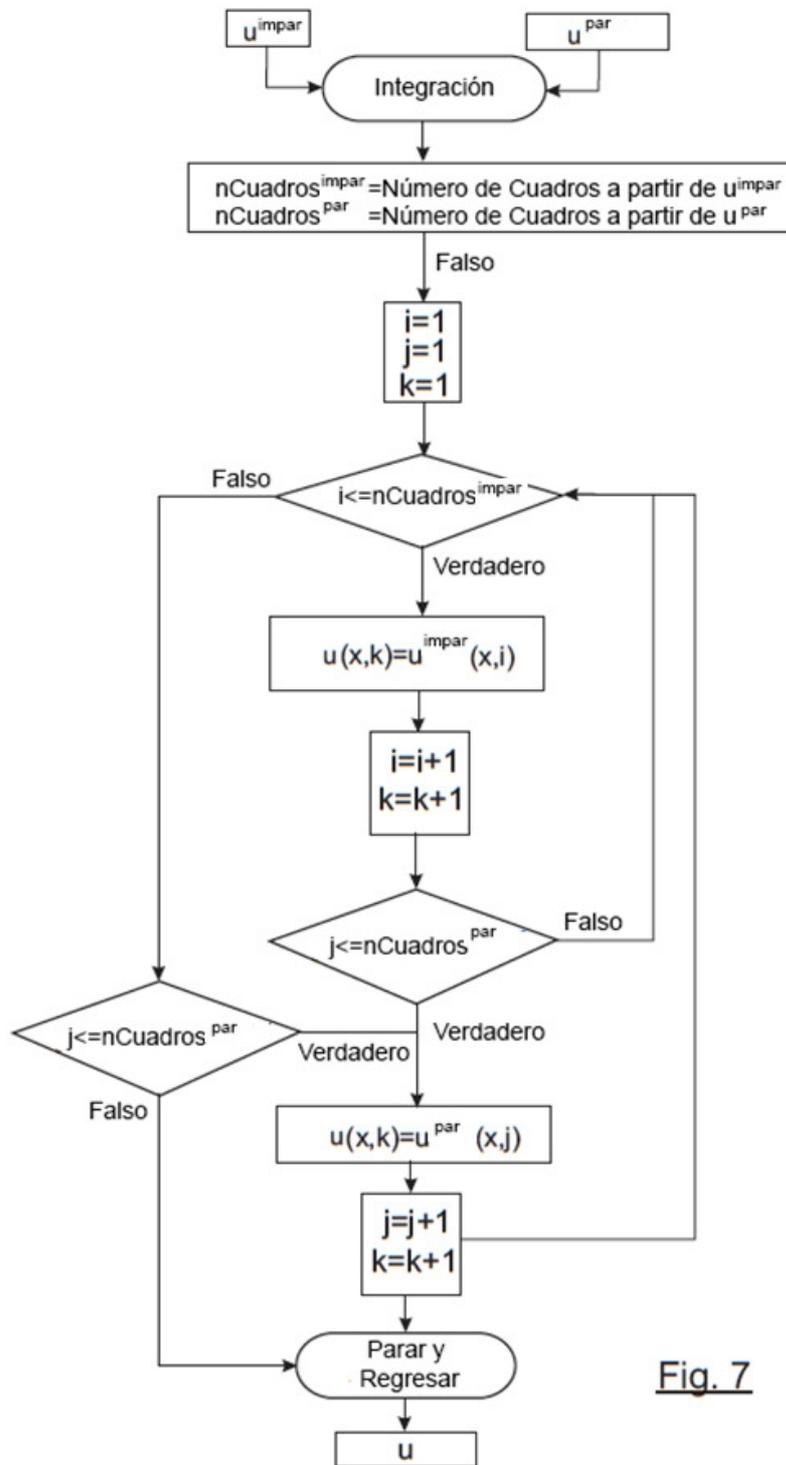


Fig. 7

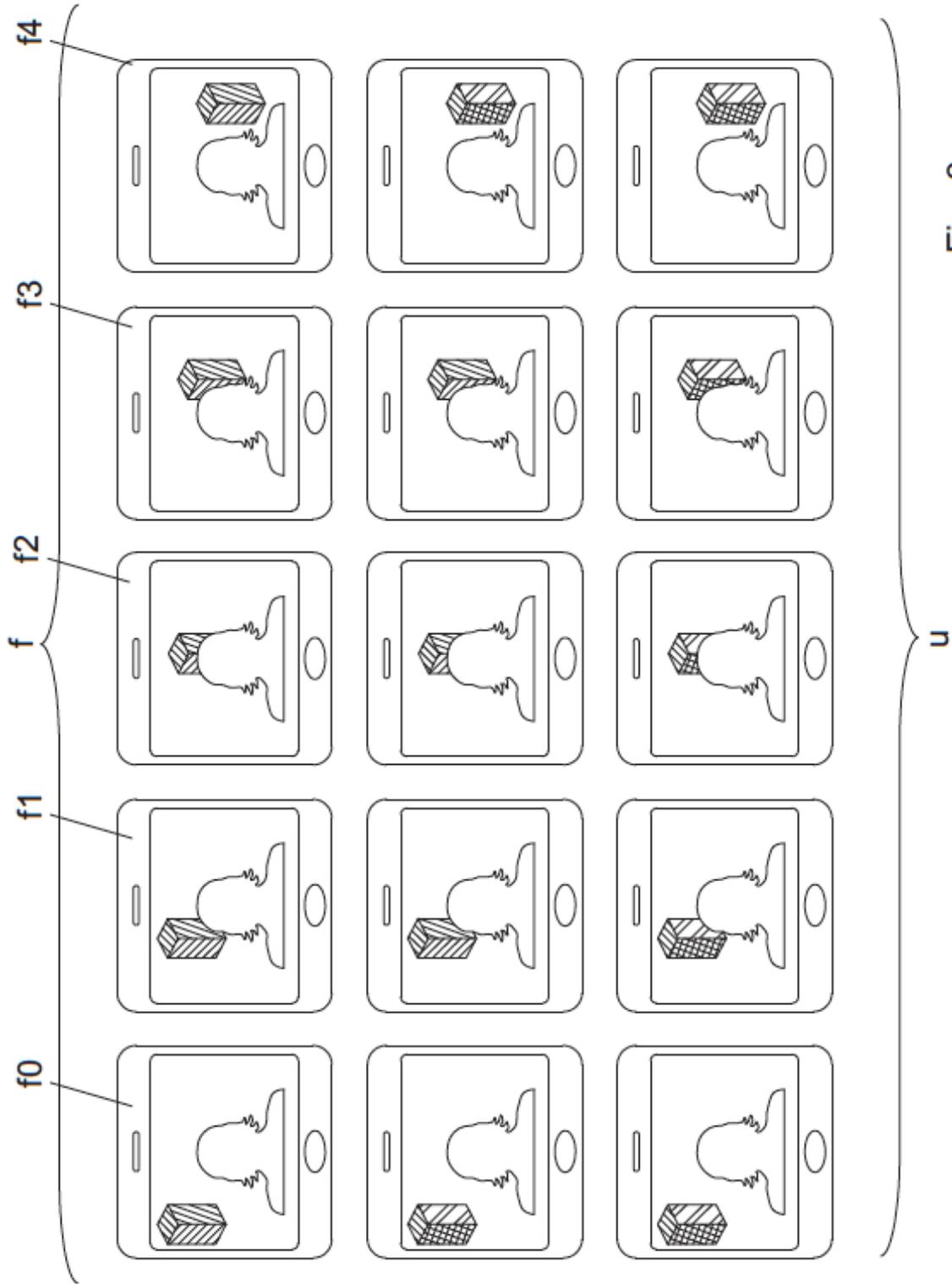


Fig. 8

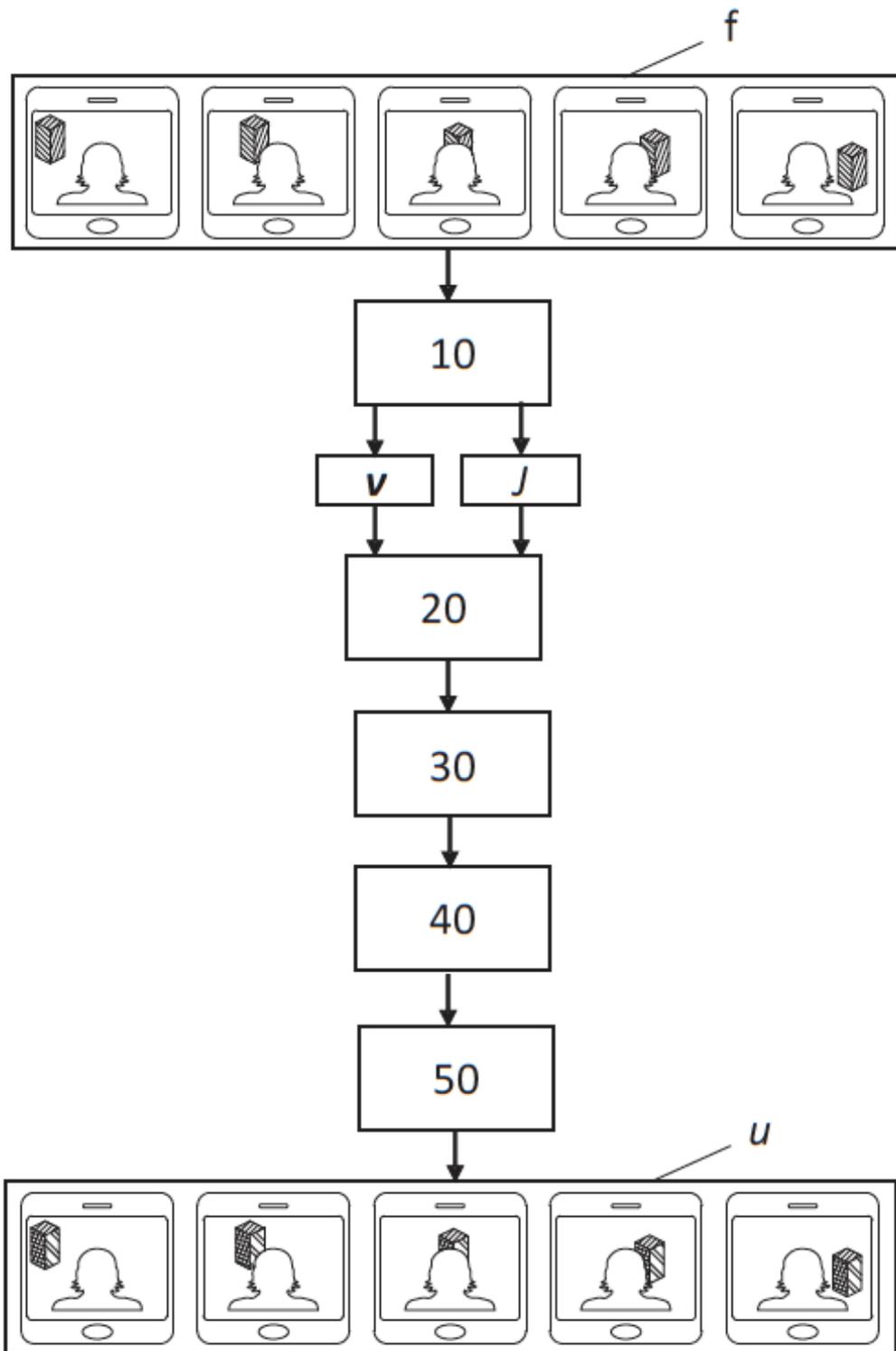


Fig. 9