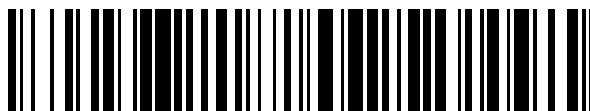


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 805 804**

51 Int. Cl.:

G06T 1/20 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **30.06.2016 PCT/IB2016/001120**

87 Fecha y número de publicación internacional: **04.01.2018 WO18002684**

96 Fecha de presentación y número de la solicitud europea: **30.06.2016 E 16757060 (5)**

97 Fecha y número de publicación de la concesión europea: **22.04.2020 EP 3479339**

54 Título: **Visualizador multimodal**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
15.02.2021

73 Titular/es:
**KEEN EYE TECHNOLOGIES (100.0%)
Incubateur Institut de la Vision, 74 rue du
Faubourg Saint-Antoine
75012 Paris, FR**

72 Inventor/es:
BERLEMONT, SYLVAIN

74 Agente/Representante:
SÁEZ MAESO, Ana

ES 2 805 804 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

Visualizador multimodal

5 Campo de la invención

La presente invención se refiere a un sistema y método de comunicación de imágenes, y más particularmente a un método de visualización y manipulación de imágenes multicanal a alta velocidad.

10 Antecedentes de la invención

Los sistemas de imágenes se están volviendo cada vez más sofisticados, con la captura de una enorme cantidad de datos a una velocidad y resolución cada vez más altas. El desafío clave para un análisis de imágenes exitoso es ver, manipular, analizar y compartir datos de imagen de una manera segura, fácil y en tiempo.

15

Una problemática de un análisis de imágenes multicanal es tener la posibilidad de mostrar una selección deseada de canales y procesar transformaciones (por ejemplo, ajuste de contraste, cambio de color ...) en cada canal por separado. Otra problemática es permitir la interpretación colaborativa de los resultados al dar a los profesionales la posibilidad de acceder a los datos mostrados simultáneamente, estén donde estén y desde cualquier dispositivo de comunicación como un ordenador, una tableta o un teléfono inteligente.

20

La invención está relacionada con un entorno integrado de soporte de proyectos completo para procesar una imagen multicanal almacenada en un servidor. Se puede acceder al entorno integrado de soporte de proyectos desde cualquier navegador web, sin la necesidad de instalar ningún complemento. El método para procesar una imagen multicanal, de acuerdo con la presente invención, debe funcionar en dispositivos con recursos limitados, como los dispositivos móviles.

25

Resumen de la invención

30

La invención se refiere a un método para procesar una imagen multicanal muy rápido que permite a los usuarios acceder simultáneamente a los datos, desde cualquier dispositivo de comunicación.

El método comprende un método de almacenamiento para transformar la imagen multicanal en varias unidades de textura 2D. Esta etapa se procesa previamente en un servidor.

35

El entorno integrado de soporte de proyectos que proporciona el método de procesamiento es accesible para una multiplicidad de usuarios desde cualquier dispositivo de comunicación, como un ordenador, una tableta o un teléfono inteligente. Por ejemplo, cuando un usuario elige una imagen multicanal almacenada en el servidor e ingresa una selección de canales, los canales relacionados se muestran en todos los dispositivos de comunicación conectados.

40

En general, para una imagen multicanal considerada, cualquiera de los usuarios, pero uno a la vez, puede elegir una selección de canales, parámetros de transformación y proyección para cada canal, en cualquier momento. Las etapas de transformación y proyección se procesan en los canales seleccionados a una velocidad muy alta gracias al procesamiento en paralelo, es decir, todos los píxeles de todos los canales seleccionados se procesan al mismo tiempo en las unidades de procesamiento de gráficos de los dispositivos conectados.

45

La invención se puede aplicar en cualquier imagen multicanal, tal como un formulario de datos de 8 bits, de dos dimensiones, de tres dimensiones o un formulario de datos de 16 bits. Los diferentes canales de la imagen pueden ser de diferentes modos de formación imágenes.

50

Se propone un método para visualizar y manipular una imagen multicanal compuesta por varios canales, cuyos canales se originan a partir de uno o varios modos de formación de imágenes, este método que comprende las etapas de:

- almacenar una imagen multicanal en una o varias unidades de textura 2D, cada una que representa un paquete de cuatro mosaicos de 8 bits correspondientes a uno o varios canales en una ubicación determinada en la imagen multicanal, cada unidad de textura 2D que es una unidad de almacenamiento de memoria de tamaño fijo en una unidad de procesamiento de gráficos,

55

- tomar en cuenta una selección de uno o varios canales de la imagen multicanal,
- tomar en cuenta los parámetros de transformación y proyección para los canales seleccionados,
- aplicar la transformación en los canales seleccionados, de acuerdo con los parámetros recibidos, todos los píxeles de los canales seleccionados que se transforman al mismo tiempo,

60

- proyectar los canales seleccionados en un espacio de color RGB, de acuerdo con los parámetros recibidos, todos los píxeles de los canales seleccionados que se proyectan al mismo tiempo.

La transformación aplicada puede ser, por ejemplo, una modificación del contraste de la imagen determinada por los parámetros de entrada.

65

La etapa de almacenamiento realizada para una imagen de tipo de datos de 8 bits, comprende las operaciones de:

- dividir cada canal de la imagen multicanal en una cuadrícula regular de mosaicos de 8 bits del mismo tamaño y del mismo tamaño que una unidad de textura 2D,
- agrupar los mosaicos de 8 bits correspondientes a una ubicación dada en la cuadrícula regular en una o varias unidades de textura 2D.

5

Cuando se realiza para una imagen de tipo de datos de 16 bits, esta etapa de almacenamiento comprende las operaciones de:

- dividir cada canal de la imagen multicanal en una cuadrícula regular de mosaicos de 16 bits del mismo tamaño y del mismo tamaño que una unidad de textura 2D,
- dividir cada mosaico de 16 bits en un par de dos mosaicos de 8 bits mediante la separación de los ocho bits más altos de los ocho bits más bajos para cada valor de píxel de 16 bits,
- agrupar los mosaicos de 8 bits correspondientes a una ubicación dada en la cuadrícula regular en una o varias unidades de textura 2D.

10

15

La etapa de transformación realizada para una imagen de tipo de datos de 8 bits, comprende las operaciones de:

- atribuir valores de intensidad del canal a los píxeles a partir de los valores almacenados en los canales de las unidades de textura 2D,
- procesar los nuevos valores de píxel de acuerdo con los parámetros de transformación de entrada.

20

La etapa de proyección realizada para una imagen de tipo de datos de 8 bits comprende la operación de procesar la proyección lineal de todos los píxeles en un espacio de color RGB, de acuerdo con los parámetros de proyección de entrada.

25

En el caso de una imagen de tipo de datos de 16 bits, esta etapa de proyección comprende las operaciones de:

- procesar la proyección lineal de todos los píxeles en un espacio de color RGB mediante el uso de los parámetros de proyección de entrada,
- convertir cada canal obtenido R, G y B en canales de 8 bits.

30

La etapa de transformación realizada para una imagen de tipo de datos de 16 bits, comprende las operaciones de:

- convertir cada par consecutivo de los canales de las unidades de textura 2D de 8 bits en valores de 16 bits,
- atribuir valores de intensidad del canal a los píxeles a partir de los valores de 16 bits,
- procesar los nuevos valores de píxel de acuerdo con los parámetros de transformación de entrada.

35

La presente invención también proporciona un sistema para procesar una imagen multicanal, la imagen multicanal compuesta de varios canales, dicho sistema que comprende un servidor que incluye:

- un módulo de conversión para convertir imágenes multicanal en una o varias unidades de textura 2D, cada una que representa un paquete de cuatro mosaicos de 8 bits correspondientes a uno o varios canales en una ubicación determinada en la imagen multicanal, cada unidad de textura 2D que es una unidad de almacenamiento de memoria de tamaño fijo en una Unidad de Procesamiento de Gráficos,
- un módulo de memoria para almacenar las unidades de textura 2D obtenidas,
- un módulo de conexión para conectar uno o varios dispositivos de comunicación, como un ordenador personal, una tableta digital o un teléfono inteligente, a una misma cuenta,
- un módulo de selección para recibir la selección de uno o varios canales de una imagen multicanal almacenada en el servidor, desde cualquiera de los dispositivos de comunicación conectados al servidor en una misma cuenta,
- un módulo de configuración para recibir los parámetros de transformación o proyección de entrada para los canales seleccionados desde cualquiera de los dispositivos de comunicación conectados al servidor en una misma cuenta,
- un módulo de procesamiento para instruir a los dispositivos conectados al servidor en una misma cuenta el procesamiento, de acuerdo con los parámetros recibidos, de una transformación y una proyección de los canales seleccionados, en un espacio de color RGB, directamente en sus Unidades de Procesamiento de Gráficos.

50

Breve descripción de las Figuras

Los objetos y ventajas anteriores y adicionales de la invención serán evidentes a partir de la descripción detallada de las modalidades preferidas, consideradas junto con los dibujos adjuntos.

55

La Figura 1 es una ilustración del sistema de comunicación global.

La Figura 2 representa un diagrama funcional de la transformación y proyección en un espacio de color RGB para un ejemplo de una imagen de tipo de datos de 8 bits.

La Figura 3 representa un diagrama funcional de la transformación y proyección en un espacio de color RGB para un ejemplo de una imagen de tipo de datos de 16 bits.

60

La Figura 4 es una ilustración de la etapa de almacenamiento para un ejemplo de una imagen de tipo de datos de 8 bits.

La Figura 5 es una ilustración de la etapa de almacenamiento para un ejemplo de una imagen de tipo de datos de 16 bits.

La Figura 6 representa un diagrama funcional de la transformación y proyección en un espacio de color RGB para un ejemplo de una imagen de tipo de datos de 8 bits.

65

La Figura 7 representa un diagrama funcional de la transformación y proyección en un espacio de color RGB para un ejemplo de una imagen de tipo de datos de 16 bits.

LISTADO 1 es un ejemplo de implementación secuencial de las etapas de transformación y proyección.

LISTADO 2 es un ejemplo de una implementación paralela de las etapas de transformación y proyección.

Descripción detallada

5 La Figura 1 representa el sistema de comunicación global. Este comprende un servidor 1 con una unidad de memoria 2 y una unidad central de procesamiento 3. La unidad de memoria 2 almacena imágenes multicanal en unidades de textura 2D. La etapa de transformar una imagen multicanal en varias unidades de textura 2D es llevada a cabo por la unidad central de procesamiento 3 del servidor 1.

10 Varios usuarios, independientemente de su número, están conectados al servidor 1 gracias a un entorno integrado de soporte de proyectos accesible desde cualquier navegador web y desde cualquier dispositivo de comunicación 4, como un teléfono inteligente, una tableta o un ordenador, como se ilustra en la Figura 1

15 Cuando uno de los usuarios envía parámetros al servidor 1 (flecha A), este ordena a los dispositivos de comunicación conectados 4 que procesen las etapas de procesamiento en sus Unidades de Procesamiento de Gráficos (flechas B). Cualquiera de los usuarios puede ingresar nuevos parámetros.

20 La alta velocidad del método de procesamiento relacionado con esta invención permite que un grupo de usuarios visualice simultáneamente los mismos datos y las modificaciones procesadas en tiempo real. En la presente descripción, la expresión de tiempo real se refiere a una operación cuyo tiempo de ejecución dura menos de 40 milisegundos.

Las modificaciones que los usuarios pueden realizar en la imagen multicanal son:

- una selección de uno o varios canales para que se muestren en las pantallas de los dispositivos,
- los parámetros de transformación de cada canal,
- los parámetros de proyección de cada canal.

25 Las etapas de transformación y proyección se llevan a cabo en la unidad de procesamiento de gráficos de cada dispositivo conectado. La Figura 2 es un diagrama funcional de las etapas de transformación y proyección en el caso de un ejemplo de imagen de tipo de datos de 8 bits. A modo de comparación, la Figura 3 ilustra estas etapas para una imagen de tipo de datos de 16 bits. Para una imagen de tipo de datos de 16 bits hay una etapa adicional de conversión de los datos en un tipo de datos de 8 bits.

30 Las etapas ilustradas en la Figura 2 y la Figura 3 son las etapas de transformación y proyección. La operación de transformación denominada H en la Figura 2 y la Figura 3 tiene dos datos de entrada para cada canal. La operación de transformación H puede ser, por ejemplo, una operación de ajuste de contraste. La siguiente descripción a veces usará el caso particular de un ajuste de contraste, pero esta invención no se limita a este tipo de transformación. Se puede aplicar cualquier tipo de transformación que implique la modificación de los valores de los píxeles (por ejemplo, enfoque de imágenes borrosas, segmentación ...).

35 Una operación de ajuste de contraste se puede expresar matemáticamente como: dada x un valor de entrada en $[0,255]$ o $[0,2^{16} - 1]$, y considerando dos parámetros ThL y ThH ,

45
$$x: \rightarrow \left\{ \begin{array}{l} 0 \text{ si } x < ThL \\ \max \text{ si } x > ThH \\ \frac{\max}{ThH-ThL} (x - ThL) \text{ de otra manera} \end{array} \right\}$$

50 En el caso de una imagen de tipo de datos de 8 bits, el valor de \max es igual a 255 y, en el caso de una imagen de tipo de datos de 16 bits, el valor de \max es igual a $2^{16} - 1$.

La operación de proyección f en la Figura 2 y la Figura 3 es una proyección lineal de las imágenes del canal en un espacio de color RGB. Se define por una matriz M cuyos valores son ajustados por un usuario. La matriz M contiene tantas líneas como canales de entrada y tres columnas.

55 Las etapas de proyección y transformación se realizan en tiempo real, lo que permite a los usuarios procesar modificaciones y visualizarlas simultáneamente.

60 Los canales de entrada de la etapa de transformación son de forma particular. Estos almacenan en el servidor 1 en unidades de textura 2D. Una unidad de textura 2D es una unidad de almacenamiento de memoria de tamaño fijo unida a unidades de procesamiento de gráficos. Una unidad de textura 2D, de acuerdo con la presente invención, es un paquete de cuatro mosaicos de 8 bits correspondientes a uno o varios canales en una ubicación dada en la imagen multicanal en cuestión.

65 La mayoría de las veces, se usa una unidad de textura 2D para mostrar una imagen RGBA, es decir, el contenido de la unidad de textura 2D se muestra directamente en la pantalla. En el uso particular relacionado con esta invención, las unidades de textura 2D se usan solo para sus funciones de almacenamiento de memoria para facilitar y mejorar la

velocidad del proceso. Las operaciones H y f presentadas anteriormente se ejecutan directamente en las unidades de textura 2D.

5 La etapa de almacenar imágenes multicanal en paquetes de mosaicos de 8 bits, de acuerdo con esta invención, se procesa previamente en la unidad central de procesamiento 3 del servidor 1. Este método de almacenamiento se ilustra en la Figura 4 para una imagen de tipo de datos de 8 bits y en la Figura 5 para una imagen de tipo de datos de 16 bits.

10 El método de almacenamiento comprende dos etapas: una etapa de formación de mosaicos y una etapa de formación de paquetes. De acuerdo con la Figura 4, la etapa de formación de mosaicos es un proceso para dividir un canal de imagen dado en una cuadrícula regular de mosaicos del mismo tamaño. Después de la etapa de formación de mosaicos, cada canal es un paquete de varios mosaicos, y cada mosaico representa una parte de la imagen multicanal. Por ejemplo, los mosaicos 100, 200, 300, 400 y 500 representan la misma ubicación en la imagen multicanal. Cada mosaico tiene el mismo tamaño en píxeles que una unidad de textura 2D, es decir, 512 x 512 píxeles.

15 La etapa de formación de paquetes consiste en agrupar cada mosaico en una ubicación dada en paquetes de cuatro canales. En el ejemplo de la Figura 4, la imagen de entrada está compuesta por cinco canales. Los mosaicos superiores izquierdos de cada canal forman un paquete (100, 200, 300 y 400) y otro paquete que contiene el mosaico 500. En este ejemplo, el número de paquetes es dos. El número de paquetes se llama K y el número de archivos se llama N en la Figura 4.

20 El número de mosaicos, paquetes y archivos depende del número de canales y del tamaño de la imagen de entrada. La regla de la etapa de formación de paquetes es minimizar el número de mosaicos al agruparlos en paquetes. Un paquete de cuatro mosaicos también se denomina unidad de textura 2D. En el ejemplo de la Figura 4, todos los mosaicos son datos de 8 bits.

25 Ahora nos referimos a la etapa de almacenamiento ilustrada en la Figura 5 para un ejemplo de una imagen multicanal de tipo de datos de 16 bits. Comprende las mismas etapas que en el caso de la imagen de tipo de datos de 8 bits, es decir, una etapa de formación de mosaicos y una etapa de formación de paquetes. Además, comprende una operación suplementaria llamada S en la Figura 5. S es una operación de división. Consiste en separar los ocho bits más altos de los ocho bits más bajos en un valor de tipo de datos de 16 bits. Esta operación se realiza después de la etapa de formación de mosaicos, en donde los mosaicos obtenidos siguen siendo de 16 bits. Por ejemplo, el mosaico superior izquierdo de 16 bits 100 se divide en dos mosaicos de 8 bits: 100h que contiene los ocho bits más altos para cada valor de píxel y 100l que contiene los ocho bits más bajos para cada valor de píxel.

35 La Figura 6 representa un diagrama funcional de las etapas de transformación y proyección en el caso del mismo ejemplo de imagen de tipo de datos de 8 bits que en la Figura 4. En particular, muestra las diferentes etapas llevadas a cabo en el primer archivo de dos unidades de textura 2D. Las mismas operaciones se llevan a cabo en todas las otras unidades de textura 2D mostradas en la Figura 4, pero no representadas aquí.

40 T1 y T2 son dos unidades de textura RGBA 2D. La operación de transformación H se realiza en todas las unidades de textura 2D en paralelo, es decir, todos los píxeles de las unidades de textura 2D se procesan al mismo tiempo. Después de la etapa de transformación, la operación de proyección f también se realiza en todos los píxeles en paralelo, lo que permite que la imagen se muestre en tiempo real (es decir, en un tiempo inferior a seis milisegundos).

45 Ahora nos referimos a la Figura 7 que representa un diagrama funcional de las etapas de transformación y proyección del mismo ejemplo de imagen de tipo de datos de 16 bits que en la Figura 5. En particular, muestra las diferentes etapas llevadas a cabo en el primer archivo de dos unidades de textura 2D. Las mismas operaciones se llevan a cabo en todas las otras unidades de textura 2D mostradas en la Figura 5, pero no representadas aquí.

50 En este diagrama hay dos operaciones suplementarias llamadas Sr y conversión de 8 bits. La operación Sr es la función inversa de la operación S ilustrada en la Figura 5. Dados dos valores de 8 bits, la función Sr recompone el valor de 16 bits. Los valores de 16 bits de los píxeles se transforman y proyectan en el espacio de color RGB directamente antes de la conversión de 8 bits en pantalla. De la misma manera que para el ejemplo de 8 bits, todos los píxeles se procesan en paralelo.

55 Para la comparación entre los algoritmos secuenciales y paralelos, nos referimos a los pseudocódigos LISTADO 1 y LISTADO 2, respectivamente. Se escriben para el caso en el cual la operación de transformación es un ajuste de contraste. De acuerdo con el LISTADO 1, el enfoque directo consiste en el procesamiento secuencial de las etapas de manera típica en una unidad central de procesamiento. Con esta implementación particular, los píxeles se procesan secuencialmente. 60 En los cambios de entrada del usuario realizados en los parámetros del vector de transformación de entrada $ThLs$ y $ThHs$, el tiempo para representar una imagen en la pantalla es demasiado grande. Como ejemplo, el tiempo promedio para llevar a cabo tres canales de 8 bits es de doscientos a trescientos milisegundos. Por lo tanto, este enfoque directo evita la actualización en tiempo real de la entrada del usuario.

65 De acuerdo con el LISTADO 2, el método de implementación de la presente invención usa la capacidad de procesamiento en paralelo de la unidad de procesamiento de gráficos para procesar todos los píxeles p en paralelo en lugar de

5 secuencialmente. Otra diferencia con el procesamiento secuencial está en las fuentes de entrada. En este método, los canales se almacenan en unidades de textura 2D, es decir, en paquetes de cuatro mosaicos, en lugar de en una memoria del sistema. Los canales de entrada se refieren al vector T de las unidades de textura 2D K. El ejemplo del LISTADO 2 es para un archivo. Los archivos N correspondientes a los canales seleccionados m se procesan en paralelo de acuerdo con esta implementación.

Para un ejemplo de acuerdo con la presente invención, el tiempo promedio para llevar a cabo cinco canales de 8 bits está comprendido entre dos y seis milisegundos. La alta velocidad de este método permite la actualización en tiempo real de la entrada del usuario.

REIVINDICACIONES

1. Método para visualizar y manipular una imagen multicanal compuesta por varios canales, cuyos canales se originan a partir de uno o varios modos de formación de imágenes, este método que comprende las etapas de:
- almacenar una imagen multicanal en una o varias unidades de textura 2D, cada una que representa un paquete de cuatro mosaicos de 8 bits correspondientes a uno o varios canales en una ubicación determinada en la imagen multicanal, cada unidad de textura 2D que es una unidad de almacenamiento de memoria de tamaño fijo en una unidad de procesamiento de gráficos,
 - tomar en cuenta una selección de uno o varios canales de la imagen multicanal,
 - tomar en cuenta los parámetros de transformación y/o proyección para los canales seleccionados,
 - aplicar la transformación en los canales seleccionados, de acuerdo con los parámetros recibidos, todos los píxeles de los canales seleccionados que se transforman al mismo tiempo,
 - proyectar los canales seleccionados en un espacio de color RGB, de acuerdo con los parámetros recibidos, todos los píxeles de los canales seleccionados que se proyectan al mismo tiempo.
2. El método, de acuerdo con la reivindicación 1, en donde la operación de transformación es una modificación del contraste de la imagen.
3. El método de acuerdo con cualquiera de las reivindicaciones 1 o 2, en donde la etapa de almacenamiento, realizada para una imagen de tipo de datos de 8 bits, comprende las operaciones de:
- dividir cada canal de la imagen multicanal en una cuadrícula regular de mosaicos de 8 bits del mismo tamaño y del mismo tamaño que una unidad de textura 2D
 - agrupar los mosaicos de 8 bits correspondientes a una ubicación dada en la cuadrícula regular en una o varias unidades de textura 2D.
4. El método de acuerdo con cualquiera de las reivindicaciones de la 1 a la 3, en donde la etapa de transformación, realizada para una imagen de tipo de datos de 8 bits, comprende las operaciones de:
- atribuir valores de intensidad del canal a los píxeles a partir de los valores almacenados en los canales de las unidades de textura 2D,
 - procesar los nuevos valores de píxel de acuerdo con los parámetros de transformación de entrada.
5. El método de acuerdo con cualquiera de las reivindicaciones de la 1 a la 4, en donde la etapa de proyección realizada para una imagen de tipo de datos de 8 bits comprende la operación de procesar la proyección lineal de todos los píxeles en un espacio de color RGB, de acuerdo con los parámetros de proyección de entrada.
6. El método de acuerdo con cualquiera de las reivindicaciones 1 o 2, en donde la etapa de almacenamiento, realizada para una imagen de tipo de datos de 16 bits, comprende las operaciones de:
- dividir cada canal de la imagen multicanal en una cuadrícula regular de mosaicos de 16 bits del mismo tamaño y del mismo tamaño que una unidad de textura 2D,
 - dividir cada mosaico de 16 bits en un par de dos mosaicos de 8 bits mediante la separación de los ocho bits más altos de los ocho bits más bajos para cada valor de píxel de 16 bits,
 - agrupar los mosaicos de 8 bits correspondientes a una ubicación dada en la cuadrícula regular en una o varias unidades de textura 2D.
7. El método de acuerdo con la reivindicación 6, en donde la etapa de transformación, realizada para una imagen de tipo de datos de 16 bits, comprende las operaciones de:
- convertir cada par consecutivo de los canales de las unidades de textura 2D de 8 bits en valores de 16 bits,
 - atribuir valores de intensidad del canal a los píxeles a partir de los valores de 16 bits,
 - procesar los nuevos valores de píxel de acuerdo con los parámetros de transformación de entrada.
8. El método de acuerdo con la reivindicación 7, en donde la etapa de proyección, realizada para una imagen de tipo de datos de 16 bits, comprende las operaciones de:
- procesar la proyección lineal de todos los píxeles en un espacio de color RGB mediante el uso de los parámetros de proyección de entrada,
 - convertir cada canal obtenido R, G y B en canales de 8 bits.
9. Un sistema para procesar una imagen multicanal, la imagen multicanal compuesta de varios canales, dicho sistema que comprende un servidor (1) que incluye:
- un módulo de conversión para convertir imágenes multicanal en una o varias unidades de textura 2D, cada una que representa un paquete de cuatro mosaicos de 8 bits correspondientes a uno o varios canales en una ubicación determinada en la imagen multicanal, cada unidad de textura 2D que es un almacenamiento de memoria de tamaño fijo en una Unidad de Procesamiento de Gráficos,
 - un módulo de memoria para almacenar las unidades de textura 2D obtenidas,
 - un módulo de conexión para conectar uno o varios dispositivos de comunicación (4), como un ordenador personal, una tableta digital o un teléfono inteligente, a una misma cuenta,

ES 2 805 804 T3

- 5
- un módulo de selección para recibir la selección de uno o varios canales de una imagen multicanal almacenada en el servidor (1), desde cualquiera de los dispositivos de comunicación conectados al servidor (1) en una misma cuenta,
 - un módulo de configuración para recibir los parámetros de transformación o proyección de entrada para los canales seleccionados desde cualquiera de los dispositivos de comunicación conectados al servidor (1) en una misma cuenta,
 - un módulo de procesamiento para instruir a los dispositivos conectados al servidor (1) en una misma cuenta el procesamiento, de acuerdo con los parámetros recibidos, de una transformación y una proyección de los canales seleccionados, en un espacio de color RGB, directamente en sus Unidades de Procesamiento de Gráficos.

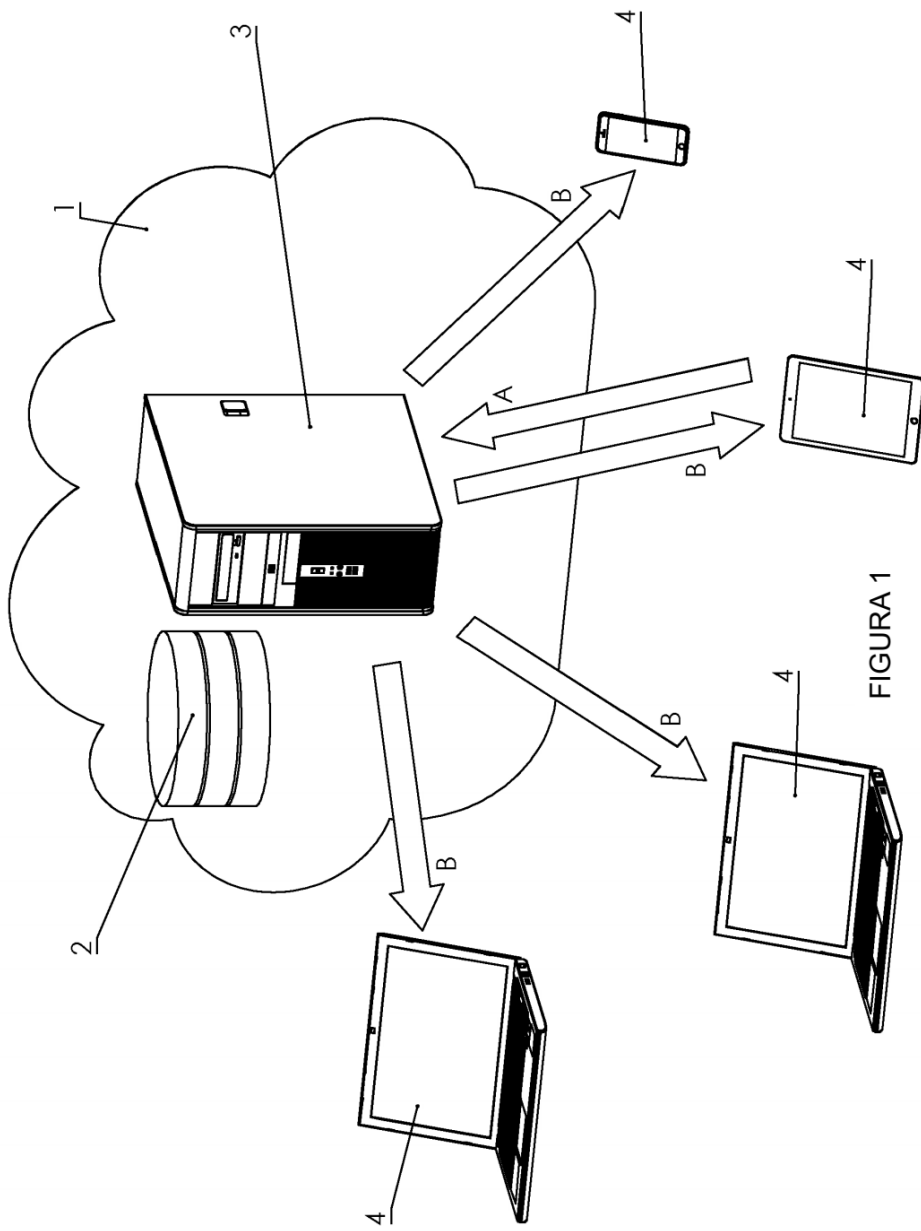


FIGURA 1

FIGURA 2

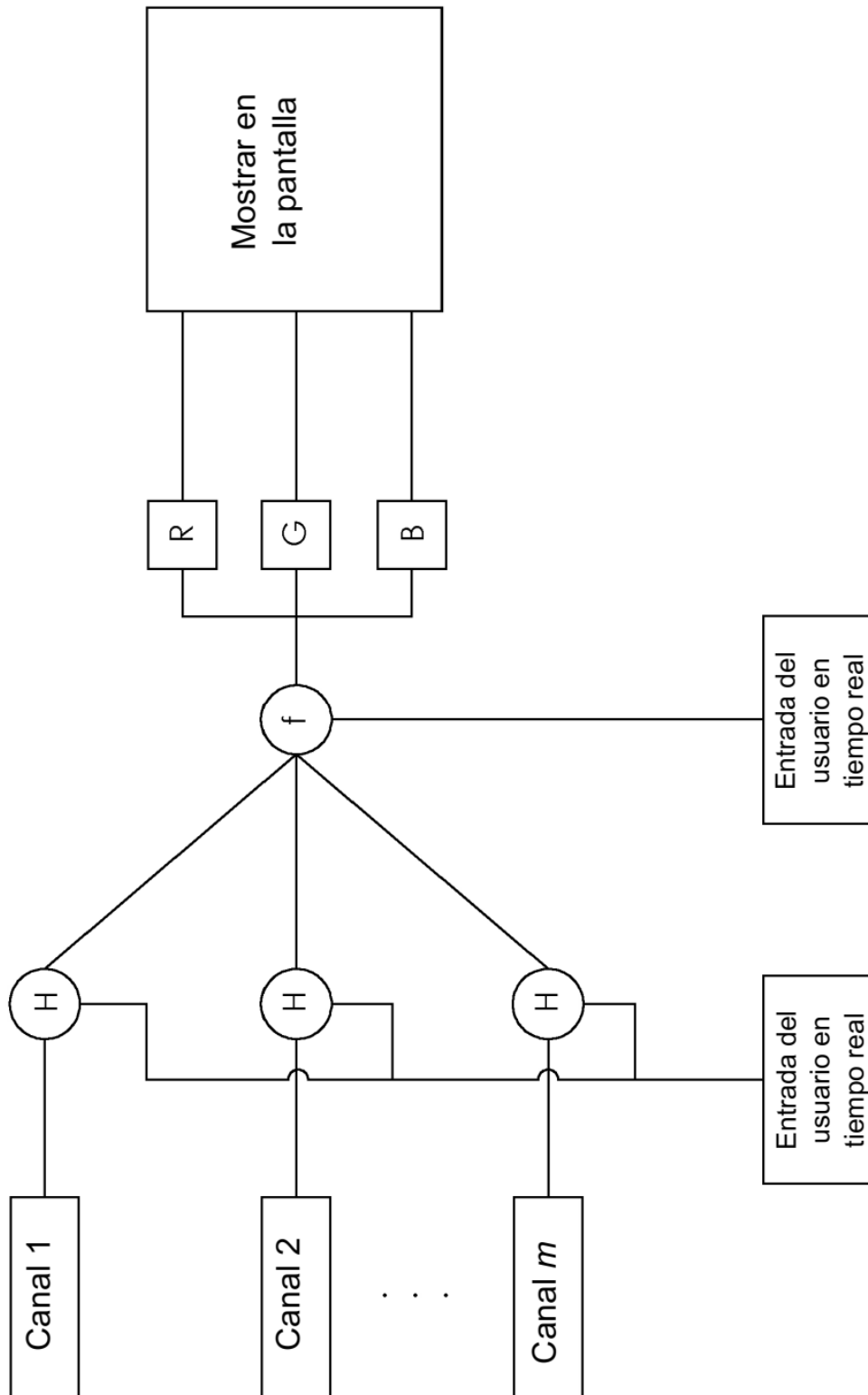
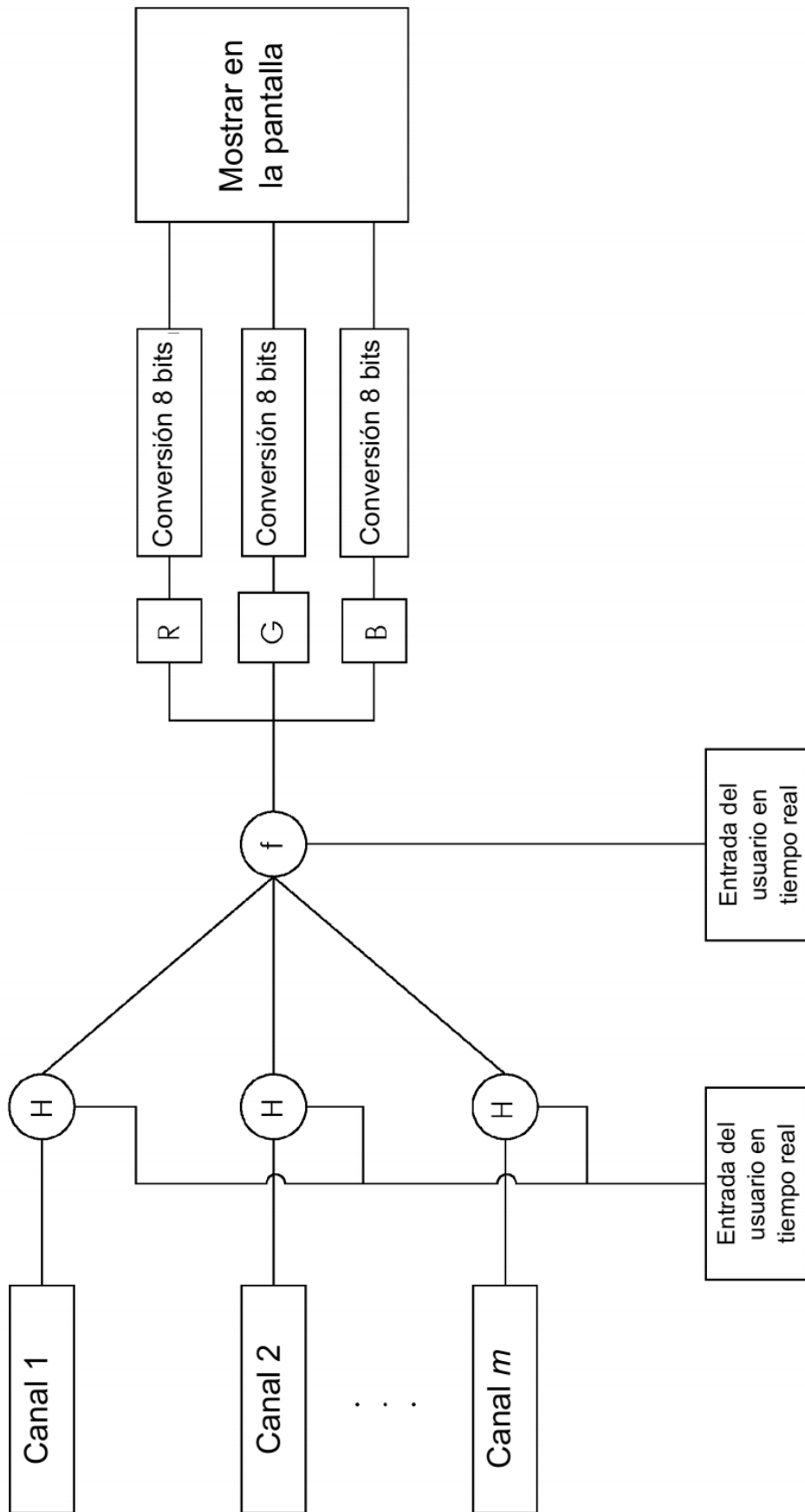


FIGURA 3



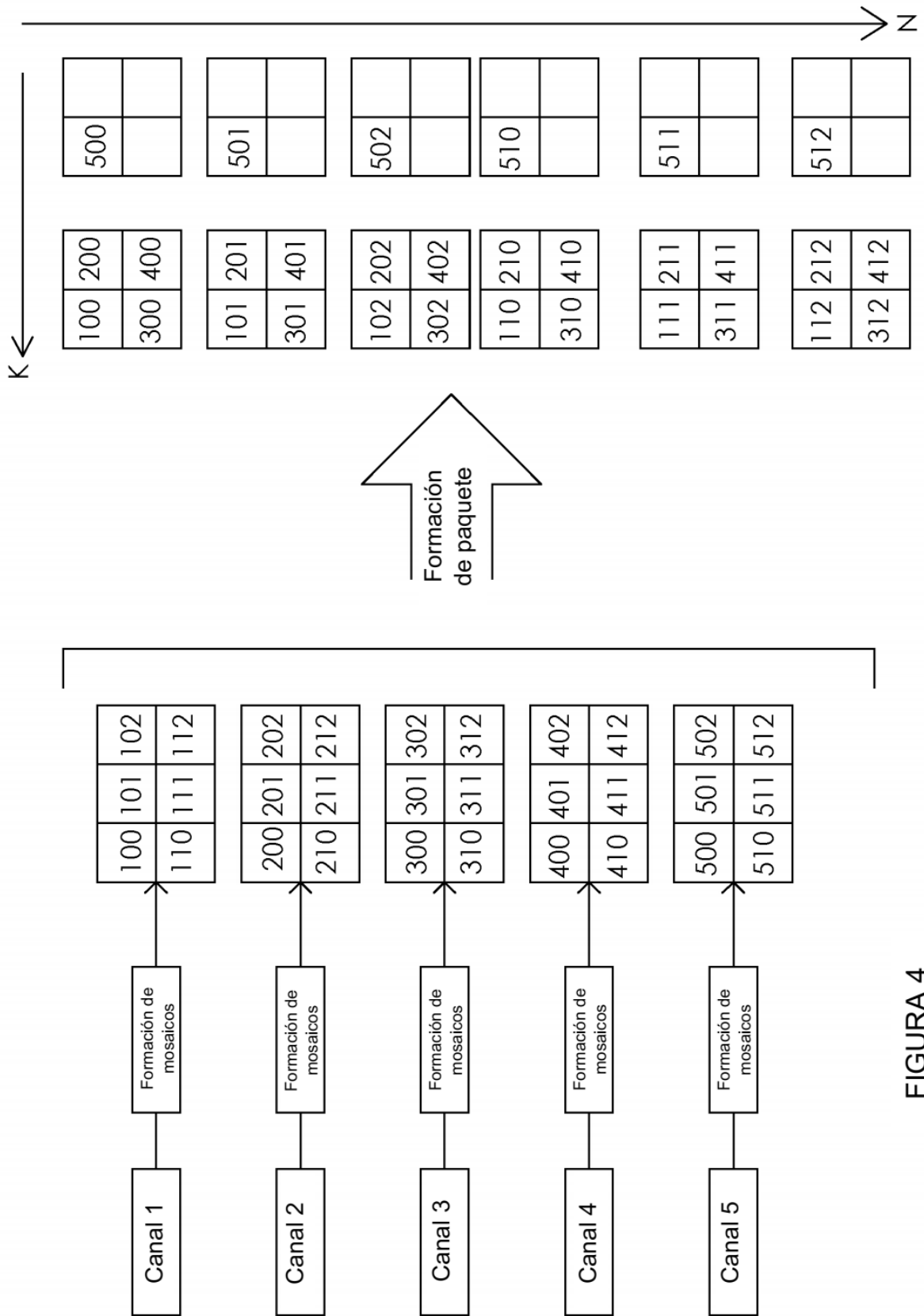


FIGURA 4

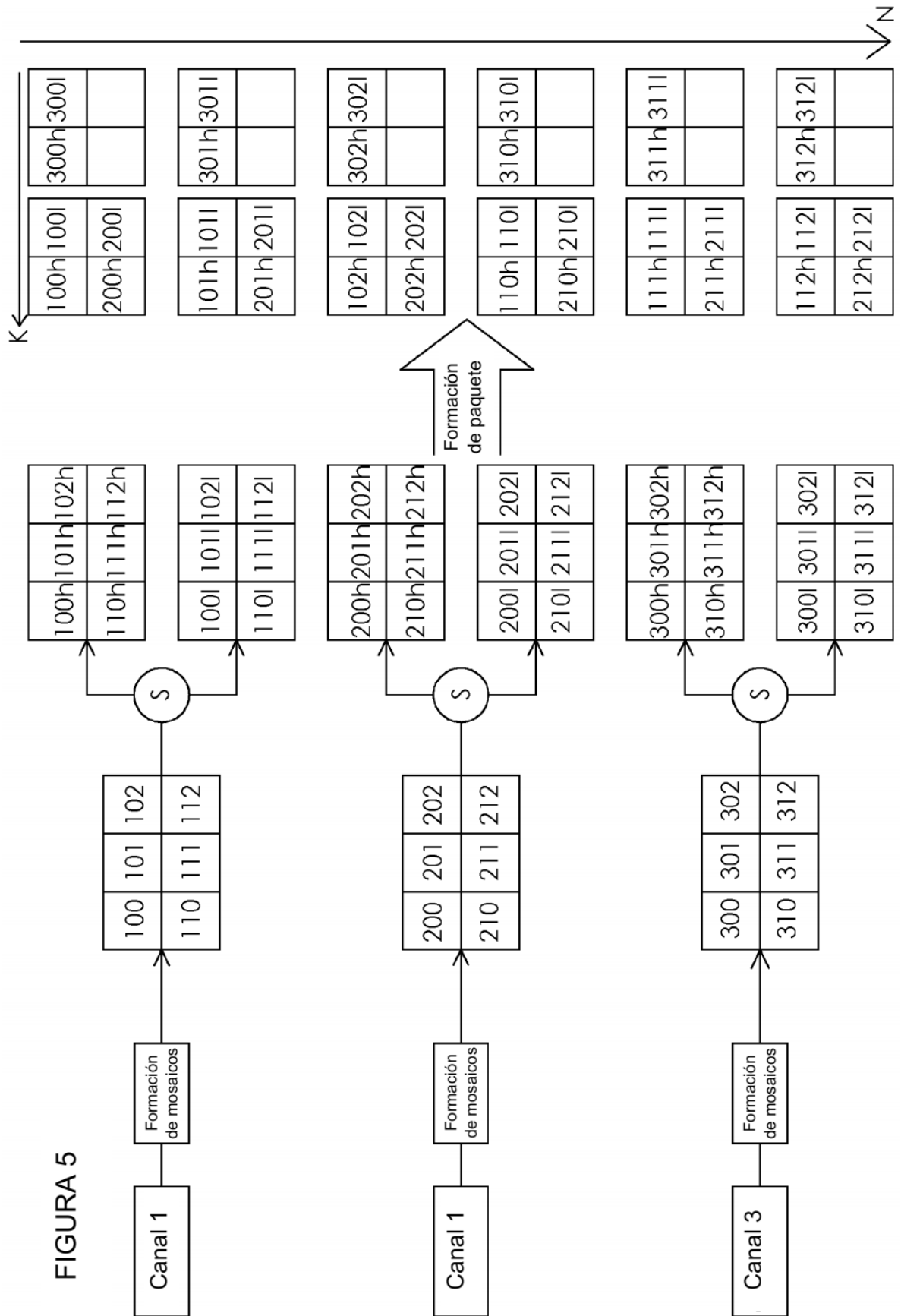


FIGURA 6

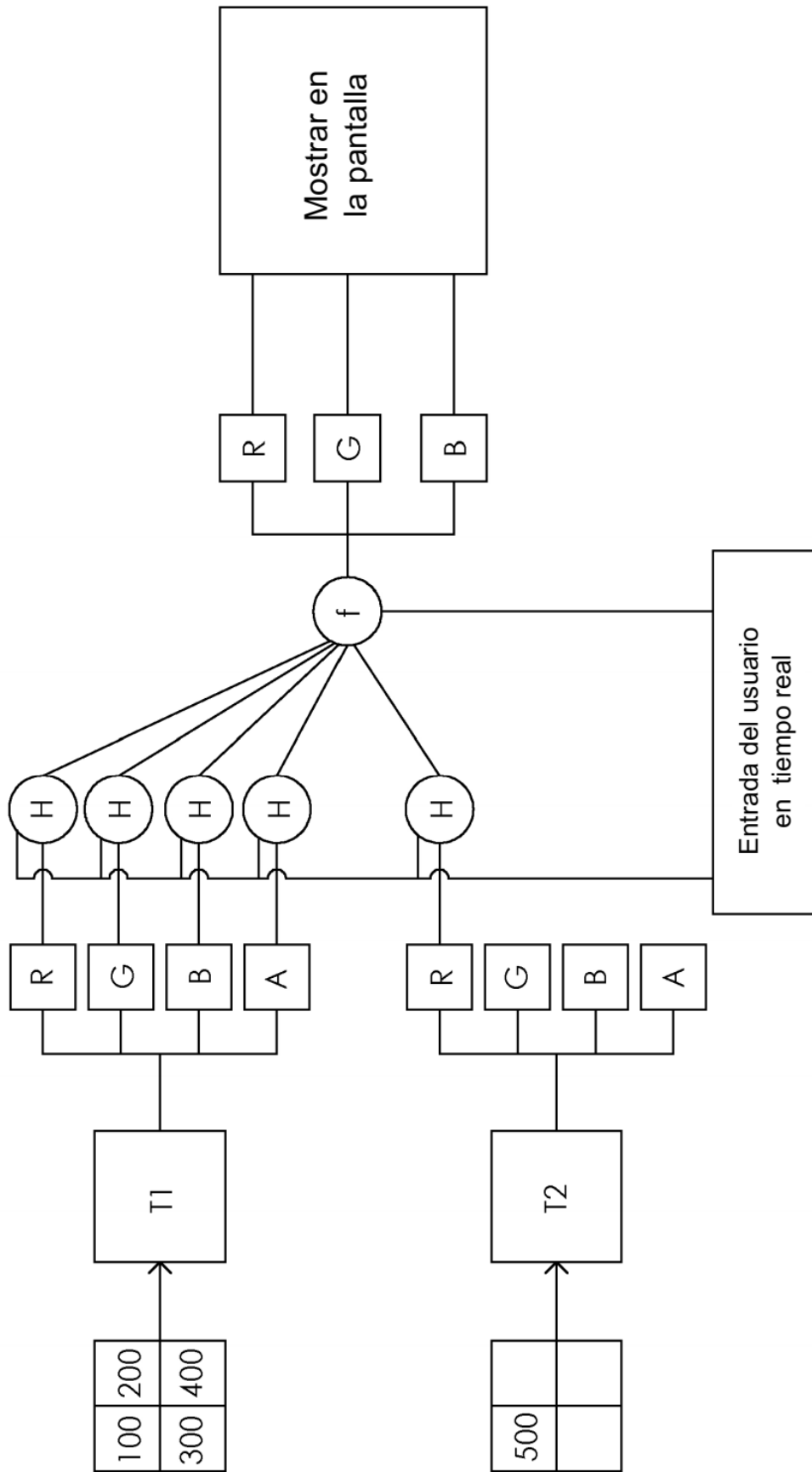
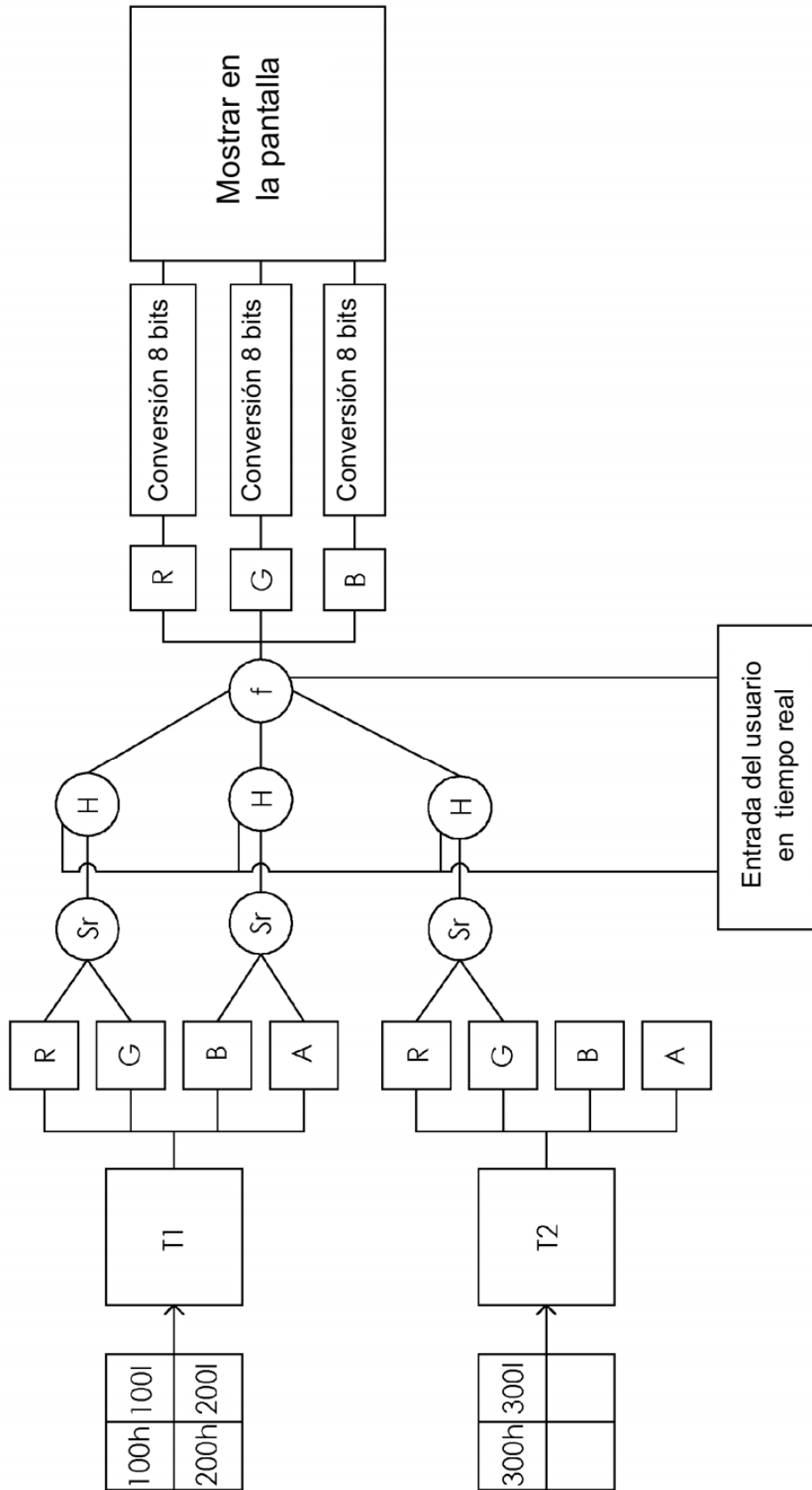


FIGURA 7



Algoritmo secuencia (Unidad de Procesamiento Central)

Entradas:

M (Matriz de proyección $m \times 3$)*ThLs* (Vector de valores m)*ThHs* (Vector de valores m)*Canales* (Vector de canales m)

Salidas:

Salida (Matriz de vectores de tres componentes que representan los valores R, G y B)

Instrucciones:

```

for each pixel  $p$  do
   $Out [1][p] = 0; Out [2][p] = 0; Out [3][p] = 0;$ 
  for  $i = 1$  to  $m$  do
     $x = Channels [i][p];$ 
    if  $x < ThLs [i]$  then
       $x = 0;$ 
    else if  $x > ThHs [i]$  then
       $x = max;$ 
    else
      
$$x = \frac{max}{(ThHs [i] - ThLs [i])} (x - ThLs [i]);$$

    end if
    for  $j = 1$  to  $3$  do
       $Out [j][p] = M[i][j].x;$ 
    end for
  end for
  display  $Out [j][p]$  at pixel  $p$ 
end for

```

LISTADO 1

Algoritmo secuencia (Unidad de Procesamiento Central)

Entradas:

M (Matriz de proyección $m \times 3$)

$ThLs$ (Vector de valores m)

$ThHs$ (Vector de valores m)

T (Vector de unidades de textura (RGBA) K)

Salidas:

Salida (Matriz de vectores de tres componentes que representan los valores R, G y B)

Instrucciones:

for all píxels p **do** //Ejecución en paralelo del siguiente programa

$Out [1][p] = 0; Out [2][p] = 0; Out [3][p] = 0;$

$offset = 0;$

for $k = 1$ to K **do**

$x [offset + 1] =$ value of p in R;

$x [offset + 2] =$ value of p in G;

$x [offset + 3] =$ value of p in B;

$x [offset + 4] =$ value of p in A;

Offset += 4;

end for

for $i = 1$ to m **do**

if $x [i] > ThLs [i]$ **then**

if $x [i] > ThHs [i]$ **then**

$x [i] = max;$

else

$x = \frac{max}{(ThHs [i] - ThLs [i])} (x [i] - ThLs [i]);$

end if

end if

$Out [1][p] = M [i][1] . x[i]$

$Out [2][p] = M [i][2] . x[i]$

$Out [3][p] = M [i][3] . x[i]$

end for

Mostrar los valores de salida en los píxeles

LISTADO 2