

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 410 304**

21 Número de solicitud: 201132008

51 Int. Cl.:

**H04N 7/26**

(2006.01)

12

## SOLICITUD DE PATENTE

A2

22 Fecha de presentación:

**14.12.2011**

43 Fecha de publicación de la solicitud:

**01.07.2013**

71 Solicitantes:

**UNIVERSIDADE DA CORUÑA (100.0%)  
A Maestranza, s/n  
15071 A Coruña ES**

72 Inventor/es:

**GULIAS FERNANDEZ, Víctor Manuel;  
TAIBO PENA, Javier;  
BARREIRO PAZ, Miguel y  
MONTERO MANSO, Pablo**

74 Agente/Representante:

**CARVAJAL Y URQUIJO, Isabel**

54 Título: **DISPOSITIVO Y SISTEMA ESCALABLE PARA LA SOBREIMPRESIÓN DE FLUJOS DE VÍDEO.**

57 Resumen:

Dispositivo y sistema escalable que permite la alteración dinámica de flujos de vídeo minimizando su recodificación para la inserción de imágenes en fotogramas, tales como los logos identificativos de canales de televisión ("mosca"), sobreimpresión de múltiples canales (multi picture-in-picture), sobreimpresión de imágenes personalizadas, entre otras. Comprende un recortador (20), un sintetizador (30), un mezclador (40) y un controlador (10). El recortador (20) recorta regiones en los fotogramas de un flujo de vídeo (50) según un patrón de recorte (52) para producir un flujo recortado (60) y flujos de recortes (70) y/o escalados (74). El sintetizador (30) sintetiza flujos de vídeo coincidentes con los recortes según un patrón de síntesis (32) para producir flujos con recortes sintetizados (80) a partir de (70, 74). El mezclador (40) mezcla los flujos (60, 80) según un patrón de mezcla (42) para producir flujos de vídeo con recortes sobreimpresos (90). El controlador (10) coordina el funcionamiento conjunto.

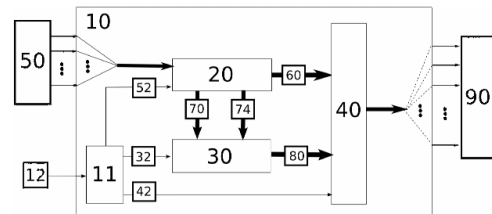


Fig. 1

## DESCRIPCIÓN

Dispositivo y sistema escalable para la sobreimpresión de flujos de vídeo.

### Sector de la Técnica

5 La presente invención pertenece al campo del tratamiento de flujos de vídeo en redes de comunicaciones, particularmente en Internet.

### Antecedentes de la invención

En el estado de la técnica se conocen diversas propuestas relacionadas con la presente invención.

10 En el documento US 6,573,945 se divulga un método para la sobreimpresión de imágenes durante el proceso de codificación. En cambio, la presente invención se realiza sobre vídeo ya codificado, minimizando la recodificación de las partes afectadas y permitiendo la generación bajo demanda de las imágenes insertadas.

El documento US 6,658,057 divulga un método para la inserción de logos en un flujo de datos. Se limita únicamente a la inserción de logos (no permite imágenes arbitrarias, ni múltiples regiones, ni combinaciones de otros flujos de vídeo), intenta reaprovechar la información de movimiento original para recodificar los bloques afectados. No está pensada para ser escalable con elevados volúmenes de canales y usuarios diferentes.

15 US 6,226,041 divulga un método para insertar logos según una técnica que se apoya en la inserción de logos sólo en fotogramas/bloques prescindibles que no son referenciados en otros fotogramas. Por tanto, esta propuesta no es general, ni es adecuada para escalar con un volumen de canales y usuarios elevado.

20 WO2001033861 propone una aproximación similar a la presente invención pero nuevamente está orientado a imágenes estáticas, no a imágenes generadas bajo demanda. Además, no permite escalar con un volumen de canales y usuarios elevado.

25 El artículo de Yu Liu; Gulling Li; Qiang Tang; JiChang Guo; , "DCT domain logo insertion of MPEG-2 transcoding," (Electrical and Computer Engineering, 2003. IEEE CCECE 2003. Canadian Conference on , vol.2, no., pp. 1219-1222 vol.2, 4-7 May 2003, doi: 10.1109/CCECE.2003.1226118), evita por completo la recodificación del flujo de vídeo comprimido mediante la inserción de imágenes en el espacio transformado, por tanto tiene una aproximación totalmente diferente.

Actualmente, se puede concluir que la sobreimpresión de imágenes en flujos de vídeo se resuelve mediante las siguientes acciones:

- Descodificar todos los flujos de entrada.
- 30 - Sobre-imponer los flujos sintéticos (esto es, flujos de vídeo cuyos fotogramas son generados por ordenador) para cada usuario sobre los fotogramas descodificados.
- Recodificar los fotogramas descodificados modificados por cada usuario.
- Enviar el vídeo comprimido codificado de forma individualizada para cada usuario.

En la práctica, esta solución presenta dos grandes dificultades que provocan su inviabilidad en sistemas reales:

- a) Coste computacional.
- 35 b) Pérdida de calidad debido a las re-codificaciones.

### Descripción de la invención

A la luz de lo anterior, sería deseable diseñar un sistema que supere las dificultades apuntadas. Por tanto, se ha de conseguir una reducción del coste computacional y evitar la pérdida de calidad resultante de la recodificación. Para ello, la presente invención emplea técnicas basadas en computación distribuida:

- 40 a) Computación distribuida para el recorte y mezcla de vídeo digital en espacio transformado, en la medida de lo posible.
- b) Computación distribuida para repartir el coste de generación de flujos sintéticos que serán sobreimpuestos.

La invención tiene, entre otras, como aplicaciones inmediatas las siguientes:

- 45 - Inserción de imágenes, como los logos identificativos de los canales de televisión ("mosca"), en tiempo real minimizando la recodificación.

- Sobreimpresión de múltiples canales (multi picture-in-picture).
- Sobreimpresión de publicidad y anuncios personalizados por usuario.
- Sobreimpresión de aplicaciones interactivas sobre canales de televisión existentes.

5 Para ello se propone un dispositivo para la sobreimpresión de flujos de vídeo que comprende un recortador, un sintetizador, un mezclador y un controlador.

El recortador recorta en los fotogramas de al menos un flujo de vídeo de entrada, al menos una región rectangular según un patrón de recorte para producir:

- Un flujo de vídeo recortado que comprende una pluralidad de bloques correspondientes a una pluralidad de fotogramas recortados.

10 - Una pluralidad de flujos de vídeo de recortes, que comprende una pluralidad de bloques correspondientes a una pluralidad de fotogramas de recortes;

- Una pluralidad de flujos de vídeo escalados, que comprende una pluralidad de bloques correspondientes al fotograma de entrada redimensionado a un tamaño distinto.

15 El recortador además asocia el flujo de vídeo recortado y los flujos de vídeo de recortes mediante la asignación de un número de secuencia a cada pareja compuesta por un fotograma recortado y los fotogramas de recorte correspondientes.

El sintetizador sintetiza dinámicamente flujos de vídeo coincidentes con los agujeros del flujo de vídeo recortado según una operación de síntesis establecida para producir flujos de vídeo con recortes sintetizados.

20 El mezclador mezcla los fotogramas con un mismo número de secuencia del flujo de vídeo recortado y de los flujos de vídeo de recortes sintetizados de acuerdo con una operación de mezcla para generar flujos de vídeo con recortes sobreimpresos.

Opcionalmente, el recortador puede rectificar la pluralidad de bloques de cada fotograma inter del flujo de vídeo de entrada recortado, si dichos bloques comprimidos tienen vectores de movimiento referenciando a bloques pertenecientes a las regiones definidas por el patrón de recorte que lo produjo.

25 Opcionalmente, el recortador puede rectificar la pluralidad de bloques de cada fotograma inter del flujo de vídeo de recortes, si dichos bloques comprimidos tienen vectores de movimiento referenciando a bloques no pertenecientes a la región descrita en el patrón de recorte que lo produjo.

30 Opcionalmente, la operación de síntesis realizada por el sintetizador para producir flujos de vídeo con recortes sintetizados puede ser escalar el recorte a un tamaño diferente, modificar la calidad del recorte, reemplazar por un color uniforme, reemplazar por una imagen estática o dinámica, reemplazar por una imagen que representa un texto en una fuente y tamaño dado, transformar un recorte (voltar horizontalmente/verticalmente, girar), modificar colores del recorte (modificar balance de color, modificar brillo y contraste, modificar tono y saturación, transformar colores, invertir colores), aplicar un filtro sobre un recorte (posterizar, saturar, enfocar, desenfocar, realzar, distorsionar, añadir luces y sombras, añadir ruido, detectar bordes), o combinación de las anteriores.

35 Opcionalmente, el mezclador comprende una memoria intermedia (o buffer) para mitigar los desfases temporales entre flujos de vídeo para resincronizar dichos flujos de vídeo utilizando el número de secuencia asignado por el recortador.

40 Es también objeto de la invención un sistema para la sobreimpresión de flujos de vídeo caracterizado por que comprende una pluralidad de dispositivos con sus elementos en cluster. Cada recortador individual de un dispositivo está configurado para recortar un subconjunto de los flujos de vídeo de entrada, de acuerdo con un subconjunto de patrones de recorte asociados para producir al menos los flujos de vídeo recortados, flujos de vídeo de recortes y flujos de vídeo escalados correspondientes a exclusivamente dicho subconjunto de flujos de vídeo de entrada.

45 Opcionalmente en el sistema anterior, los sintetizadores de cada dispositivo están distribuidos en cluster y cada sintetizador individual está configurado para recibir los flujos de vídeo de recortes y flujos de vídeo escalados correspondientes con un subconjunto de flujos de vídeo de entrada para producir un subconjunto de flujos de vídeo con recortes sintetizados de acuerdo con unos patrones de síntesis asociados.

50 Opcionalmente en el sistema anterior, los mezcladores de cada dispositivo están distribuidos en cluster y cada mezclador individual está configurado para recibir los flujos de vídeo recortados y los flujos con recortes sintetizados correspondientes con un subconjunto de flujos de vídeo de entrada para producir un subconjunto de flujos de vídeo con recortes sobreimpresos.

Opcionalmente el sistema puede incluir un controlador configurado para recibir un patrón de superposición asociado a un usuario para construir los correspondientes patrones de recorte, síntesis y mezcla, con los que configurar el recortador, el sintetizador y el mezclador, respectivamente, para enviar la pluralidad de flujos de vídeo con recortes superpuestos por una red de telecomunicaciones (por ejemplo Internet) a una pluralidad de direcciones destino asignadas a dichos usuarios.

**Descripción de los dibujos**

A continuación se pasa a describir de manera muy breve una serie de dibujos que ayudan a comprender mejor la invención y que se relacionan expresamente con una realización de dicha invención que se presenta como un ejemplo no limitativo de ésta.

- 10 Fig. 1 describe un ejemplo de realización de un dispositivo para la sobre-impresión que comprende un recortador (20), un sintetizador (30), un mezclador (40) y un controlador (10)
- Fig. 2 describe un ejemplo concreto formado por dos flujos de vídeo IP (50) a la entrada de un recortador (20) parametrizado con tres patrones de recorte (52), un sintetizador (30) parametrizado con cinco patrones de síntesis (32) y un mezclador (40) parametrizado con cuatro patrones de mezcla (42).
- 15 Fig. 3 describe un recortador en cluster que comprende varios recortadores (20) acoplados.
- Fig. 4 muestra un diagrama de flujo de las principales etapas llevadas a cabo en un recortador (20).
- Fig. 5 describe en detalle la etapa de recortado y rectificado (21), en la que se recorta (56) el fotograma actual comprimido (51) y éste se rectifica (57) usando un fotograma de referencia descomprimido (53).
- 20 Fig. 6 describe en detalle la etapa de escalado (25), en la que se escala (28) el fotograma actual descomprimido (54) y, opcionalmente, se codifica (26) a algún formato de vídeo (por ejemplo, MPEG-2 ISO 13818).

**Realización preferente de la invención**

Sin carácter limitativo, el sistema propuesto por la presente invención puede implementarse mediante cuatro componentes:

- 25 a) Un recortador (20) diseñado para procesar unos flujos de vídeo de entrada (50) según unos patrones de recorte (52) dados y para generar como resultado diferentes flujos independientes de vídeo de salida (60,70), preferentemente: flujos de vídeo de entrada recortados (60) (el vídeo original con agujeros), flujos de vídeo con los recortes (70) (el contenido de los agujeros) y flujos de vídeo de entrada escalados (74).
- 30 b) Un sintetizador (30) diseñado para generar unos flujos de vídeo con recortes sintetizados (80) (vídeo de tamaño coincidente con los recortes) según unos patrones de síntesis (32) a partir de los flujos de vídeo de entrada recortados (70) y flujos de vídeo de entrada escalados (74).
- c) Un mezclador (40) diseñado para mezclar el flujo de vídeo de entrada recortado (60) y flujos de vídeo con recortes sintetizados (80) según unos patrones de mezcla (42) para construir un flujo de vídeo de entrada con recortes superpuestos (90) (el vídeo original con agujeros adecuadamente rellenos con recortes sintetizados) que será enviado de forma individual a cada usuario.
- 35 d) Un controlador (10) diseñado para coordinar el funcionamiento conjunto de recortador (20), sintetizador (30) y mezclador (40).

A continuación se describen en detalle cada uno.

**a) Recortador**

40 El recortador (20) recibe  $n$  flujos de vídeo IP (50)  $c_1, c_2, \dots, c_n$  y, para cada flujo  $c_i$  recibe una serie de  $k_i$  patrones de recorte (52)  $\{pr_{i1}, pr_{i2}, \dots, pr_{ik}\}_i$ , y genera 3 tipos de flujos de salida diferentes:

$t$ : (RECORTADO) flujos de vídeo IP de entrada recortados (60), esto es, los vídeos de entrada en los que se han suprimido apropiadamente una o más regiones rectangulares disjuntas.

$r$ : (RECORTE) flujos de vídeo IP con los recortes (70) de la entrada, esto es, los flujos correspondientes a cada una de las regiones suprimidas apropiadamente del flujo de vídeo de entrada.

45  $s$ : (ESCALADO) flujos de vídeo IP de entrada escalados (74), esto es, los vídeos de entrada redimensionados normalmente a una resolución o escala menor.

Cada patrón de recorte (52)  $pr_{ij}$  es un conjunto de regiones rectangulares, no solapadas y alineadas al tamaño de bloque utilizado por el estándar de vídeo utilizado, identificadas por el par (esquina superior izquierda, esquina inferior derecha) para el flujo de entrada (50)  $c_i$ .

**i) Flujos de vídeo IP de entrada recortados (60), t:**

5 Cada flujo de vídeo con un canal de entrada recortado  $t_{ij}$  se construye a partir de un flujo de vídeo de entrada (50)  $c_i$  en el cual se eliminan adecuadamente (ver PROCESO DE RECORTADO, ESCALADO Y ENVÍO) todos los bloques incluidos en todas las regiones rectangulares definidas por el patrón de recorte (52) específico  $pr_{ij}$ , obteniendo un flujo de vídeo (60) en el que todos sus fotogramas tienen agujeros consistentes con el patrón de recorte.

10 Si existen  $n$  flujos de entrada y, para cada flujo  $c_i$ , existen  $k_i$  patrones de recorte (52) distintos, se generarán  $\sum_{i=1..n}(k_i)$  flujos distintos de vídeo (60), identificados como  $t_{11}, t_{12}, \dots, t_{1k}, t_{21}, t_{22}, \dots, t_{2k}, \dots, t_{n1}, t_{n2}, \dots, t_{nk}$ . Junto con cada fotograma de cada flujo de vídeo se envía un número de secuencia único del fotograma (61,71,73) que será utilizados posteriormente por el sintetizador (30) y el mezclador (40) para asociar adecuadamente dicho fotograma recortado (61) con los recortes sintetizados (81) para este mismo fotograma (ver SINTETIZADOR y MEZCLADOR).

15 Cada flujo de vídeo IP de entrada recortado (60) se envía por la red a múltiples destinos usando algún procedimiento de envío multipunto, por ejemplo enviando el flujo a un grupo multicast específico.

**ii) Flujos de vídeo IP con los recortes (70) de la entrada, r:**

20 Los distintos patrones de recorte (52) para un mismo flujo de vídeo IP de entrada definen regiones a recortar, algunas de las cuales pueden ser coincidentes en distintos patrones de recorte (52). Denominamos  $R_i$  (55) al conjunto de esas regiones para el flujo de vídeo IP de entrada  $c_i$ ; es necesario generar un único flujo de vídeo con el recorte de la entrada (70) para cada región en  $R_i$  (55).

25 Para cada flujo de vídeo IP de entrada  $c_i$  (50) y para cada una de las regiones rectangulares del conjunto  $R_i$  (55) definidas en algún patrón de recorte  $pr_{ij}$  (52) para dicho canal de entrada se generará un flujo de vídeo IP (70)  $r_{ik}$  con el recorte del flujo de entrada correspondiente a la región rectangular  $k$  perteneciente a  $R_i$ . Dicho recorte  $r_{ik}$  se construye eliminando adecuadamente (ver PROCESO DE RECORTADO, ESCALADO Y ENVÍO) todos los bloques NO incluidos en la región rectangular. Si  $R_{pr}(pr_{ij})$  son las regiones definidas por el patrón de recorte  $pr_{ij}$ ,  $R_i$  es el conjunto  $\bigcup_{i=1..k} (R_{pr}(pr_{ij}))$ .

Para cada flujo de vídeo IP de entrada (50)  $c_i$  se generarán  $|R_i|$  flujos IP con los recortes de la entrada (70), identificados como  $r_{i1}, r_{i2}, \dots, r_{ij} \in R_i$ , generándose en total  $\sum_{i=1..n}(|R_i|)$  flujos.

30 Junto con cada fotograma (71) de cada flujo de vídeo con los recortes (70) de la entrada se envía el número de secuencia único del fotograma (51) que será utilizado posteriormente por el sintetizador (30) y el mezclador (40) para asociar adecuadamente el fotograma recortado (61) con los recortes sintetizados (81) para un mismo fotograma (51) (ver SINTETIZADOR y MEZCLADOR).

35 Cada flujo de vídeo IP con los recortes (70) de la entrada se envía por la red a múltiples destinos usando algún procedimiento de envío multipunto, por ejemplo enviando el flujo a un grupo multicast específico. El flujo de vídeo IP con los recortes (70) de la entrada puede estar comprimido según algún estándar o descomprimido.

**iii) Flujos de vídeo IP de entrada escalados, s:**

40 Cada flujo de vídeo de entrada escalado (74) se construye redimensionando los fotogramas (51) de un flujo de vídeo IP (50) de la entrada a  $Q$  tamaños de fotograma distintos (73). Si existen  $n$  flujos de entrada, se generarán  $Q \times n$  flujos distintos de vídeo escalado (74), identificados como  $s_{11}, s_{12}, \dots, s_{1Q}, s_{21}, s_{22}, \dots, s_{2Q}, \dots, s_{n1}, s_{n2}, \dots, s_{nQ}$ .

Cada flujo de vídeo IP de entrada escalado (74) se envía por la red a múltiples destinos usando algún procedimiento de envío multipunto, por ejemplo enviando el flujo a un grupo multicast específico. El flujo de vídeo IP con los recortes (70) de la entrada puede estar comprimido según algún estándar o descomprimido (ver PROCESO DE RECORTADO, ESCALADO Y ENVÍO).

**45 Proceso de recortado, escalado y envío**

Aclarar previamente que dentro de un fotograma inter (P-frame, B-frame) puede haber bloques predictivos (inter) o no predictivos (intra). De otra parte, un fotograma intra (I-frame) sólo tiene bloques intra.

Por cada flujo de entrada (50)  $c_i$  se dispone de un número de secuencia y se sigue el siguiente procedimiento:

- i) A partir del flujo de entrada (50)  $c_i$  se extrae el siguiente fotograma (58), el fotograma actual comprimido (51), y se incrementa el número de secuencia.
- ii) Se descomprime (23) el fotograma actual (51), obteniendo el fotograma actual descomprimido (54).
- iii) Se almacena (24) el fotograma actual descomprimido (54).
- 5 iv) Se aplica la etapa de escalado (25) para cada una de las  $Q$  calidades  $q$  :
  - 1. se escala (28) el fotograma actual descomprimido (54) a la resolución correspondiente a  $q$ .
  - 2. si el flujo se envía comprimido, se comprime el fotograma escalado (26) según el estándar utilizado (por ejemplo, MPEG-2 ISO 13818).
  - 10 3. se encapsula y envía el fotograma escalado (73) (comprimido, si es el caso) en el flujo de vídeo escalado  $s_{iq}$  (74).
- v) Se aplica la etapa de recortado y rectificado (21) sobre el fotograma actual comprimido (51) para cada patrón de recorte (52)  $pr_{ij}$  con  $j = 1..k_i$  :
  - 1. se recorta (56) el fotograma actual (51), eliminando los bloques comprimidos que están dentro de alguna de las regiones rectangulares definidas por el patrón de recorte (52)  $pr_{ij}$ .
  - 15 2. si el fotograma actual (51) es un fotograma inter, se rectifican (57) los bloques comprimidos con vectores de movimiento apuntando a algún bloque en las regiones rectangulares definidas por el patrón de recorte (52)  $pr_{ij}$  (ver PROCESO DE RECTIFICACIÓN DE VECTORES DE MOVIMIENTO).
  - 3. se encapsulan y envían los bloques del conjunto en el flujo (60)  $t_{ij}$ , junto con el número de secuencia.
- vi) Se calculan los recortes a obtener (27) obteniendo  $R_i$  (55).
- 20 vii) Se aplica la etapa de recortado y rectificado (21) sobre el fotograma actual comprimido (51) para cada región  $k$  en  $R_i$  (55):
  - 1. se recorta (56) el fotograma actual (51), eliminando los bloques comprimidos que están fuera de la región  $k$ .
  - 25 2. si el fotograma actual (51) es un fotograma inter y los recortes se envían de forma comprimida, se rectifican (57) los bloques comprimidos de  $k$  que tengan vectores de movimiento apuntando a algún bloque fuera de  $k$  (ver PROCESO DE RECTIFICACIÓN DE VECTORES DE MOVIMIENTO).
  - 3. se encapsulan y envían los bloques del conjunto en el flujo (70)  $r_{ik}$ , junto con el número de secuencia.
- viii) ir a i) para continuar procesando el resto del flujo de vídeo IP (50) de entrada.

**Proceso de rectificación de vectores de movimiento**

- 30 Para evitar que en el tratamiento por separado del fotograma recortado y sus recortes la información de movimiento sea incorrecta (lo cual se traduciría en artefactos visuales, esto es, ruido apreciable por el usuario en la descodificación del vídeo mezclado resultante), los vectores de movimiento asociados a cada bloque comprimido deben ser analizados y rectificados (57) en el caso de que referencien a un bloque externo al conjunto de bloques a rectificar. A la entrada de este proceso tenemos una colección de bloques comprimidos, previamente recortados a partir del fotograma actual (51), a rectificar. En función del tipo de bloque comprimido se actúa del modo siguiente:
  - 35 i) Bloques intra: el bloque comprimido está auto-contenido, no existen referencias de movimiento con respecto a otros bloques. No es necesario ningún procesamiento adicional.
  - ii) Bloques inter: los vectores de movimiento hacen referencia a una región de un fotograma anterior (en P-frames y B-frames) o posterior (sólo en B-frames).
    - 40 i.a) En el caso de una referencia anterior (en P-frames y B-frames), si la región referenciada está íntegramente contenida dentro del conjunto de bloques a rectificar, entonces no es necesario ningún procesamiento adicional. En caso contrario, el bloque inter será RECTIFICADO, siendo reemplazado por un nuevo bloque resultado de comprimir como bloque intra, según el estándar utilizado (por ejemplo, MPEG-2 ISO 13818), la región correspondiente a partir de un fotograma anterior de referencia descomprimido (53), previamente almacenado (24).
    - 45

- i.b) En el caso de referencias posteriores (en B-frames), los estándares de vídeo (por ejemplo, el estándar MPEG-2 ISO 13818) garantizan que los fotogramas se reordenan dentro de los flujos de tal forma que los fotogramas referenciados siempre son enviados antes en el flujo de vídeo, siendo responsabilidad de los descodificadores el adecuado buffering y reordenación en la reproducción. Por lo tanto, el procesamiento es idéntico al presentado en (i.a), pero utilizando un fotograma que fue descomprimido (53) y almacenado (24) previamente, aunque en la secuencia lógica de reproducción sea correspondiente a un fotograma posterior.

5  
10 Opcionalmente, y para evitar la propagación de errores residuales en la rectificación (traducida en pequeños artefactos) a lo largo de una secuencia de bloques inter, si la región referenciada ha sido parcialmente rectificada desde el último fotograma intra, se rectificará como si referenciase a una región que no está íntegramente contenida en los bloques comprimidos a rectificar.

**Recortador en cluster**

15 Un recortador en cluster es un recortador cuya responsabilidad se reparte entre varios sub-recortadores, esto es, varios recortadores individuales (20) por razones de escalabilidad. Los sub-recortadores (20) pueden recibir sus entradas de redes distintas, pero sus salidas deben estar en la misma red que el sintetizador (30) (o los sub-sintetizadores, en el caso de un sintetizador en cluster), para los flujos de vídeo IP con los recortes (70) de la entrada y los flujos de vídeo IP de entrada escalados (74), y del mezclador (40) (o los sub-mezcladores, en el caso de un mezclador en cluster), para los flujos de vídeo IP con la entrada recortada (60). Los sub-recortadores procesan sus entradas de la misma forma que un recortador individual (20).

20 Todos los sub-recortadores, de forma conjunta, funcionan como si fueran un recortador único. Los  $n$  flujos de vídeo de entrada (50) se distribuyen sobre  $x$  sub-recortadores, encargados de procesar a lo sumo  $\text{ceil}(n/x)$ , donde  $\text{ceil}(x)$  es la función que calcula el menor entero mayor o igual que  $x$ . Cada sub-recortador  $i$  recibe los flujos de entrada  $(i-1)*\text{ceil}(n/x)+1$  a  $i*\text{ceil}(n/x)$ , con la excepción de  $i=x$ , en cuyo caso recibe los flujos de entrada  $(x-1)*\text{ceil}(n/x)+1$  a  $n$ .

25 Cada sub-recortador  $i$  recibe únicamente los patrones de recorte (52) correspondientes a los, a lo sumo,  $\text{ceil}(n/x)$  flujos de vídeo IP de entrada (50) que debe procesar.

Cada sub-recortador  $i$  produce los flujos (70)  $r$ , (74)  $s$  y (60)  $t$  correspondientes a los, a lo sumo,  $\text{ceil}(n/x)$  flujos de vídeo IP de entrada (50) que debe procesar, entregándolos en los grupos multicast correspondientes.

La salida conjunta del recortador distribuido es el agregado de las salidas individuales de cada sub-recortador.

**b) Sintetizador**

30 Si  $|R_i|$  es el número de regiones rectangulares distintas definidas en los patrones de recorte (52)  $pr_{ij}$  que afectan al flujo de vídeo de entrada (50)  $c_i$ , el sintetizador recibe:

c.1)  $\sum_{i=1..n}(|R_i|)$  flujos de vídeo con los flujos de vídeo IP con los recortes de la entrada (70), identificados como  $r_{11}, r_{12}, \dots, r_{1|R_1|}, r_{21}, r_{22}, \dots, r_{2|R_2|}, \dots, r_{n1}, r_{n2}, \dots, r_{n|R_n|}$  producidos por el recortador (20). Dichos flujos

35 incluyen el número de secuencia del fotograma (51) dentro del flujo y la información del tipo de fotograma a utilizar (I-frame, P-frame, B-frame).

c.2)  $n \times Q$  flujos de vídeo de entrada escalados (74), identificados como  $s_{1q}, s_{2q}, \dots, s_{nq}$ , para  $q = 1..Q$ , donde  $Q$  es el número de escalados distintos utilizados en el sistema.

40 c.3)  $Z$  patrones de síntesis (32), cada uno asociando un flujo de vídeo IP con los recortes de entrada (70), una operación de síntesis (33) concreta para modificar dicho recorte y un subconjunto de los flujos de entrada escalados (74).

45 Un patrón de síntesis (32) define la forma en que se genera un recorte sintetizado (80), como por ejemplo, mezclando el recorte de la entrada con una imagen estática (mosca), sustituyendo el recorte de la entrada por otra entrada distinta escalada (picture-in-picture), aplicar un filtro al recorte de la entrada (por ejemplo, invertir el vídeo), etc.; dado que las operaciones pueden necesitar para la generación del recorte sintetizado el recorte de entrada original, otros vídeos de entrada escalados, esta información forma parte del patrón de síntesis (32).

50 Cada patrón de síntesis (32)  $ps_z$  es un conjunto de tuplas (flujo de vídeo IP de recortes de entrada (70), operación de síntesis (33), conjunto de flujos de vídeo IP de entrada escalado (74)) que asocia un flujo de vídeo IP de recortes de la entrada (70)  $r_z$  con una operación de síntesis (33) concreta a utilizar ( $ops_z$ ) y un subconjunto de los flujos de vídeo IP de entrada escalados (74) ( $S_z$ ) de la forma  $(r_z, ops_z, S_z)$ . En particular,  $S_z$  puede ser el conjunto vacío, en aquellas operaciones de síntesis (33) que no requieran ningún flujo de vídeo IP de entrada escalados (74).

El sintetizador (30) genera  $Z$  flujos de vídeo IP con recortes sintetizados (80), uno por cada patrón de síntesis (32) distinto, identificados como  $z_1, z_2, \dots, z_z$ , exactamente de las mismas dimensiones que el flujo de vídeo IP con los

recortes de la entrada (70) asociado en el patrón de síntesis (32). Cada flujo de vídeo IP con recortes sintetizados (80) se envía por la red a múltiples destinos usando algún procedimiento de envío multipunto, por ejemplo enviando el flujo a un grupo multicast específico para ese flujo de recorte sintetizado (80). Junto con cada flujo de vídeo se incluye el número de secuencia de fotograma que será utilizado posteriormente por el mezclador (40).

5 La operación de síntesis (33)  $ops_z$  determina cómo se construye el flujo de recorte sintetizado (80)  $z_z$  a ser generado por el sintetizador (30), es decir, la forma en que se construye un flujo con una región de cada fotograma (81) y que sustituirá al recorte (71) en los agujeros del flujo de vídeo recortado (61) a la salida del mezclador, distinguiendo:

10 b.1) flujo de vídeo generado (80) a partir de los flujos de vídeo IP de entrada escalados  $S_z$  (74). De esta forma, por ejemplo, se podría sustituir el recorte por el flujo de vídeo escalado de otro canal para definir una superposición de un canal sobre otro (picture-in-picture).

b.2) flujo de vídeo generado (80) a partir de flujo de vídeo IP con recorte de la entrada (74)  $r_z$ . De esta forma, por ejemplo, se podría sustituir el recorte por la aplicación de un filtro sobre el recorte, por ejemplo para resaltar en vídeo inverso una región del fotograma.

15 b.3) flujo de vídeo generado (80) independientemente de b.1 y b.2, por ejemplo, a partir de vídeo almacenado, imágenes estáticas, imagen generada u otros flujos de vídeo distintos de los contemplados en el resto del sistema. De esta forma, por ejemplo, se podría reemplazar el recorte (71) por una imagen que representase un anuncio publicitario (banner).

20 b.4) combinaciones de los anteriores mediante operaciones gráficas (transparencias, rotaciones, deformaciones, etc). De esta forma, por ejemplo, se podría definir una mosca no rectangular, estableciendo como fondo el recorte y sobreimponiéndole una imagen estática combinadas con transparencias.

**Sintetizador en cluster**

25 Un sintetizador en cluster es un sintetizador (30) cuya responsabilidad se reparte entre varios sub-sintetizadores por razones de escalabilidad. Los sub-sintetizadores reciben todas sus entradas de una misma red y su salida debe estar en la misma red que el mezclador (40) (o los sub-mezcladores, en el caso de un mezclador en cluster). Los sub-sintetizadores procesan sus entradas de la misma forma que un sintetizador individual (30).

Todos los sub-sintetizadores, de forma conjunta, funcionan como si fueran un sintetizador único. El sintetizador se particiona en  $p$  sub-sintetizadores, cada uno encargado de generar un subconjunto de los flujos de vídeo IP con recortes sintetizados. Cada sub-sintetizador recibe un subconjunto disjuncto de los patrones de síntesis (32).

30 Cada sub-sintetizador recibe todos los flujos de entrada escalados (74)  $s$  que estén referenciados por algún patrón de síntesis (32) asignado al sub-sintetizador. Para ello, el sub-sintetizador se suscribe a un punto específico de recepción multipunto para los flujos de entrada escalado de su interés, por ejemplo suscribiéndose a los grupos multicast correspondientes a cada flujo de entrada escalado presente en los patrones de síntesis (32) asignados al sub-sintetizador, esto es,  $\bigcup_{z \in \text{asignado}(S_z)} s$ .

35 Cada sub-sintetizador recibe los recortes de flujos de entrada (70)  $r$  que estén referenciados por algún patrón de síntesis (32) asignado al sub-sintetizador. Para ello, el sub-sintetizador se suscribe a un punto específico de recepción multipunto para los flujos de entrada escalado de su interés, por ejemplo suscribiéndose a los grupos multicast correspondientes a cada flujo de vídeo IP de recortes (70) de entrada presente en los patrones de síntesis (32) asignados al sub-sintetizador, esto es,  $\bigcup_{z \in \text{asignado}(\{r_z\})} r$ .

La salida conjunta del sintetizador en cluster es el agregado de las salidas individuales de cada sub-sintetizador.

40 **c) Mezclador**

Si  $k_i$  es el número de patrones de recorte (52) definidos para cada flujo (50)  $c_i$  de los  $n$  flujos de entrada y  $m$  es el número de usuarios a los que da servicio el sistema, el mezclador (40) recibe:

45 c.1)  $\sum_{i=1..n}(k_i)$  flujos de vídeo IP con los flujos de entrada recortados (60), identificados como  $t_{11}, t_{12}, \dots, t_{1k}, t_{21}, t_{22}, \dots, t_{2k}, \dots, t_{n1}, t_{n2}, \dots, t_{nk}$ , producidos por el recortador (20). Dichos flujos incluyen el número de secuencia del fotograma dentro del flujo, añadido por el recortador.

c.2)  $p$  flujos de vídeo IP con recortes sintetizados (80), identificados como  $z_1, z_2, \dots, z_p$ . Dichos flujos incluyen el número de secuencia del fotograma dentro del flujo, añadido por el sintetizador (30) a partir de la información proporcionada por el recortador (20).

c.3)  $m$  patrones de mezcla (42) que asocian flujos de vídeo IP de entrada recortados (60) con los flujos de



vídeo IP con recortes sintetizados (80), para construir un flujo de vídeo IP de la entrada con recortes sobreimpresos para cada uno de los  $m$  usuarios del sistema.

5 Cada patrón de mezcla (42)  $pm_x$  es un conjunto de tuplas (flujo de vídeo IP de entrada recortado (60), secuencia de flujos de vídeo IP de recortes sintetizado (80), dirección destino del usuario) que asocian cada flujo de entrada recortado con una secuencia  $Z_x = \{Z_{x_1}, Z_{x_2}, \dots, Z_{x_{|pr_{ij}}}\}$  de los flujos de vídeo IP con recortes sintetizados (80), de la forma  $(t_{ij}, Z_x, D_x)$ . El número de elementos en la secuencia  $Z_x$  coincide con  $|pr_{ij}|$ , el número de regiones definidas en el patrón de recorte  $pr_{ij}$ . Adicionalmente, las dimensiones de cada una de las regiones definidas en  $Z_x$  coinciden con cada una de las regiones del patrón de recorte  $pr_{ij}$ . El flujo resultante (90) se entrega en la dirección destino  $D_x$ .

10 El mezclador produce  $m$  flujos (90) combinando los fotogramas del flujo de vídeo IP de entrada recortado (60) y de los flujos de vídeo IP de recortes sintetizado (80) que coinciden en número de secuencia (generados originalmente por el recortador (20) y propagados por el sintetizador (30)), utilizando buffers intermedios para mitigar los desfases temporales entre flujos. Estos flujos con vídeo de entrada al que se le superponen recortes sintéticos (90) se envían por la red a las direcciones destino  $D_1, D_2, \dots, D_m$ , por ejemplo una dirección unicast y puerto UDP, uno para cada uno de los  $m$  usuarios.

### 15 Mezclador en cluster

Un mezclador en cluster es un mezclador (40) cuya responsabilidad se reparte entre varios sub-mezcladores por razones de escalabilidad. Los sub-mezcladores reciben los flujos de vídeo de entrada recortado (60) de la misma red que las salidas de los sub-recortadores, y los flujos de vídeo con recortes sintetizados (80) de la misma red que las salidas de los sub-sintetizadores; la salida de cada sub-mezclador puede estar en redes diferentes. Los sub-mezcladores procesan sus entradas de la misma forma que un mezclador individual (40).

20 Todos los sub-mezcladores, de forma conjunta, funcionan como si fueran un mezclador único. El mezclador se divide en  $y$  sub-mezcladores encargados de generar el flujo de vídeo IP de la entrada con recortes sobreimpresos (90) para, a lo sumo,  $ceil(m/y)$  usuarios. Cada sub-mezclador  $i$  recibe los patrones de mezcla (42) correspondientes a los usuarios  $(i-1)*ceil(m/y) + 1$  a  $i*ceil(m/y)$  con la excepción de  $i=y$ , en cuyo caso recibe los patrones de mezcla  $(y-1)*ceil(m/y) + 1$  a  $m$ .

25 Cada sub-mezclador recibe únicamente los flujos de vídeo IP con recortes sintetizados (80)  $z$  relevantes para los patrones de mezcla (42) recibidos en el sub-mezclador. Para ello, el sub-mezclador se suscribe a un punto específico de recepción multipunto para los flujos de vídeo con recortes sintetizados (80) de su interés, por ejemplo suscribiéndose a los grupos multicast correspondientes a cada flujo de vídeo con recortes sintetizados presente en los patrones de mezcla asignados al sub-mezclador, esto es,  $union_{pm \text{ asignado}(Z_x)}$ .

30 Cada sub-mezclador recibe únicamente, a lo sumo,  $ceil(m/y)$  flujos de vídeo IP de entrada recortados (60) correspondientes a los patrones de mezcla (42) recibidos en el sub-mezclador. Para ello, el sub-sintetizador se suscribe a un punto específico de recepción multipunto para los flujos de vídeo IP de entrada recortados (60) de su interés, por ejemplo suscribiéndose a los grupos multicast correspondientes a cada flujo de vídeo IP de entrada recortado presente en los patrones de mezcla asignados al sub-mezclador, esto es,  $union_{pm \text{ asignado}(\{t_{ij}\})}$ .

35 Cada sub-mezclador produce los, a lo sumo,  $ceil(m/y)$  flujos de vídeo para cada uno de los, a lo sumo,  $ceil(m/y)$  usuarios con flujos de vídeo IP de entrada sobreimpreso con recortes sintéticos (90).

La salida conjunta del mezclador en cluster es el agregado de las salidas individuales de cada sub-mezclador.

### d) Controlador

40 El controlador (10) está a cargo de coordinar el funcionamiento conjunto de recortador (20), sintetizador (30) y mezclador (40). El controlador (10) recibe, para cada uno de los  $m$  usuarios en el sistema, un patrón de superposición (12)  $po_k = (c_i, pr_k, Ops_k, D_k)$ , que asocia un flujo de entrada (50)  $c_i$ , un patrón de recorte (52)  $pr_k$  para  $c_i$ , y una secuencia de pares (operación de síntesis (33), conjunto de flujos de entrada escalados (74))  $Ops_k$ , para cada una de las regiones rectangulares de  $pr_k$ . Con esta información el sistema genera un flujo de vídeo IP con la entrada  $c_i$  sobre-impuesta (90) de acuerdo al patrón de sobre-imposición (12) y la envía a la dirección destino  $D_k$ .

45 El proceso de admisión (11) seguido por el controlador (10) para incorporar un nuevo usuario  $k$  es el siguiente:

- i) Recibir el patrón de superposición (12)  $po_k = (c_i, pr_k, Ops_k, D_k)$ .
- ii) Determinar si en el recortador (20) existe un  $pr_{ij} = pr_k$  para  $c_i$ ; si no existe entonces añade al recortador (20) el nuevo patrón de recorte (52)  $pr_{ij} = pr_k$  para el flujo de entrada (50)  $c_i$ . En caso de que el recortador (20) esté distribuido en cluster, el controlador (20) se comunica exclusivamente con el sub-recortador a cargo del flujo de entrada (50)  $c_i$ .

- iii) Para cada uno de los elementos en  $Ops_k = \{(ops_{k1}, S_{k1}), (ops_{k2}, S_{k2}), \dots, (ops_{k|pr_{ij}|}, S_{k|pr_{ij}|})\}$ , añadir en el sintetizador (40), de no existir previamente, el nuevo patrón de síntesis (32)  $ps_{kl} = (r_{il}, ops_{kl}, S_{kl})$ , donde  $r_{il}$  es el flujo de vídeo con recortes de la entrada (60) correspondiente a cada región  $l$  definida por  $pr_{ij}$  para  $l = 1..|pr_{ij}|$ . Sea  $U_k$  la secuencia de  $|pr_{ij}|$  flujos de vídeo IP con recortes sintetizados (80) resultantes de los nuevos patrones de síntesis (32) añadidos. En caso de que el sintetizador (30) esté distribuido en cluster, el controlador (10) asigna los patrones de síntesis (32) a los sub-sintetizadores más descargados de acuerdo con alguna política definida (por ejemplo, la máquina con menor ocupación de CPU), no necesariamente a los mismos sub-sintetizadores.
- iv) Se añade en el mezclador (30) un nuevo patrón de mezcla (42)  $pm_k = (t_{ij}, U_k, D_k)$ . En caso de que el mezclador esté distribuido en cluster, el controlador se comunica exclusivamente con el sub-mezclador a cargo del usuario  $k$ .

REFERENCIAS NUMÉRICAS

- 10 Controlador
- 11 Proceso de admisión de un usuario
- 15 12 Patrón de sobreimposicion
- 20 Recortador
- 21 Etapa de recortado y rectificado
- 23 Descodificación
- 24 Almacenar fotograma de referencia descomprimido
- 20 25 Etapa de escalado
- 26 Codificar fotograma descomprimido escalado
- 27 Cálculo de  $R_i$
- 28 Escalar fotograma descomprimido
- 30 Sintetizador
- 25 32 Patrón de síntesis
- 33 Operación de síntesis
- 40 Mezclador
- 42 Patrón de mezcla
- 50 Flujo de vídeo de entrada
- 30 51 Fotograma del vídeo de entrada
- 52 Patrón de recorte
- 53 Fotograma de referencia descomprimido
- 54 Fotograma actual descomprimido
- 55 Unión de patrones de recorte,  $R_i$
- 35 56 Recorte de fotograma comprimido
- 57 Rectificación de bloques comprimidos
- 58 Extracción del fotograma actual comprimido
- 60 Flujo de vídeo recortado
- 61 Fotograma del flujo de vídeo recortado
- 40 70 Flujo de vídeo de recortes

- 71 Fotograma del flujo de vídeo de recortes
- 73 Fotograma del flujo de vídeo escalado
- 74 Flujo de vídeo escalado
- 80 Flujo de vídeo de recortes sintetizados
- 5 81 Fotograma del flujo de vídeo de recortes sintetizados
- 90 Flujo de vídeo con recortes sobreimpresos
- 91 Fotograma del flujo de vídeo con recortes sobreimpresos

**REIVINDICACIONES**

1. Dispositivo escalable para la sobreimpresión de flujos de vídeo caracterizado por que comprende:
- un recortador (20) configurado para recortar en los fotogramas (51) de al menos un flujo de vídeo de entrada (50), al menos una región rectangular según un patrón de recorte (52) para producir:
    - 5 - un flujo de vídeo recortado (60) que comprende una pluralidad de bloques correspondientes a una pluralidad de fotogramas recortados (61);
    - una pluralidad de flujos de vídeo de recortes (70), que comprende una pluralidad de bloques correspondientes a una pluralidad de fotogramas de recortes (71);
    - y una pluralidad de flujos de vídeo escalados (74), que comprende una pluralidad de bloques correspondientes al fotograma de entrada (51) redimensionado a un tamaño distinto;
  - 10 donde el recortador (20) está además configurado para asociar ambos flujos de vídeo (60,70) mediante la asignación de un número de secuencia a cada pareja compuesta por un fotograma recortado (61) y los fotogramas de recorte (71) correspondientes;
    - 15 - un sintetizador (30) configurado para sintetizar dinámicamente flujos de vídeo coincidentes con los agujeros del flujo de vídeo recortado (60) según una operación de síntesis establecida (32) para producir flujos de vídeo con recortes sintetizados (80);
    - un mezclador (40) configurado para mezclar los fotogramas (61,81) con un mismo número de secuencia del flujo de vídeo recortado (60) y de los flujos de vídeo de recortes sintetizados (80) de acuerdo con una operación de mezcla (42) para generar flujos de vídeo con recortes sobreimpresos (90).
  - 20 **2.** Dispositivo de acuerdo con la reivindicación 1, caracterizado por que el recortador (20) está configurado además para rectificar la pluralidad de bloques de cada fotograma inter del flujo de vídeo de entrada recortado (60), si dichos bloques comprimidos tienen vectores de movimiento referenciando a bloques pertenecientes a las regiones definidas por el patrón de recorte (52) que lo produjo.
  - 25 **3.** Dispositivo de acuerdo con la reivindicación 2, caracterizado por que el recortador (20) está configurado para rectificar la pluralidad de bloques de cada fotograma inter del flujo de vídeo de recortes (70), si dichos bloques comprimidos tienen vectores de movimiento referenciando a bloques no pertenecientes a la región descrita en el patrón de recorte (52) que lo produjo.
  - 30 **4.** Dispositivo de acuerdo con una cualquiera de las reivindicaciones anteriores, caracterizado por que la operación de síntesis (33) realizada por el sintetizador (30) para producir flujos de vídeo con recortes sintetizados (80) comprende al menos una de las siguientes:
    - escalar el recorte a un tamaño diferente,
    - modificar la calidad del recorte,
    - reemplazar por un color uniforme
    - reemplazar por una imagen estática o dinámica,
    - reemplazar por una imagen que representa un texto en una fuente, color y tamaño dado,
    - 35 - transformar un recorte: voltear horizontalmente/verticalmente, girar.
    - modificar colores del recorte: modificar balance de color, modificar brillo y contraste, modificar tono y saturación, transformar colores, invertir colores.
    - aplicar un filtro sobre un recorte: posterizar, saturar, enfocar, desenfocar, realzar, distorsionar, añadir luces y sombras, añadir ruido, detectar bordes.
  - 40 o combinación de las anteriores.
  - 5.** Dispositivo de acuerdo con una cualquiera de las reivindicaciones anteriores, caracterizado por que el mezclador (40) comprende una memoria intermedia (buffer) para mitigar los desfases temporales entre flujos de vídeo (60,80) y por que el mezclador (40) está configurado para resincronizar dichos flujos de vídeo (60,80) utilizando el número de secuencia asignado por el recortador (20).
  - 45 **6.** Sistema escalable para la sobreimpresión de flujos de vídeo caracterizado por que comprende una pluralidad de dispositivos de acuerdo con una cualquiera de las reivindicaciones anteriores 1 a 5, donde los recortadores de cada dispositivo están distribuidos en cluster y cada recortador individual (20) de un dispositivo está configurado para

recortar un subconjunto de los flujos de vídeo de entrada (50), de acuerdo con un subconjunto de patrones de recorte (52) asociados para producir al menos los flujos de vídeo recortados (60), flujos de vídeo de recortes (70) y flujos de vídeo escalados (74) correspondientes a exclusivamente dicho subconjunto de flujos de vídeo de entrada (50).

- 5     **7.** Sistema de acuerdo con la reivindicación 6, caracterizado por que los sintetizadores (30) de cada dispositivo están distribuidos en cluster y cada sintetizador individual (30) está configurado para recibir los flujos de vídeo de recortes (70) y flujos de vídeo escalados (74) correspondientes con un subconjunto de flujos de vídeo de entrada (50) para producir un subconjunto de flujos de vídeo con recortes sintetizados (80) de acuerdo con unos patrones de síntesis (32) asociados.
- 10    **8.** Sistema de acuerdo con la reivindicación 7, caracterizado por que los mezcladores (40) de cada dispositivo están distribuidos en cluster y cada mezclador individual (40) está configurado para recibir los flujos de vídeo recortados (60) y los flujos con recortes sintetizados (80) correspondientes con un subconjunto de flujos de vídeo de entrada (50) para producir un subconjunto de flujos de vídeo con recortes sobreimpresos (90) destinados a un subconjunto de usuarios.
- 15    **9.** Sistema de acuerdo con la reivindicación 8, caracterizado por que comprende un controlador configurado para recibir un patrón de superposición (12) asociado a un usuario para construir los correspondientes patrones de recorte (52), síntesis (32) y mezcla (42), con los que configurar el recortador (20), el sintetizador (30) y el mezclador (40), respectivamente, para enviar la pluralidad de flujos de vídeo con recortes sobreimpresos (90) por una red de telecomunicaciones a una pluralidad de direcciones destino asignadas a dichos usuarios.
- 20    **10.** Sistema de acuerdo con la reivindicación 9, caracterizado por que la red de telecomunicaciones es Internet.