(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 934 641**

(51) Int. CI.:

G06N 3/08 (2006.01)
H04N 19/117 (2014.01)
H04N 19/124 (2014.01)
H04N 19/154 (2014.01)
H04N 19/59 (2014.01)
H04N 19/172 (2014.01)
H04N 19/179 (2014.01)
H04N 19/85 (2014.01)
G06N 3/04 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **30.09.2020 E 20199342 (5)**

(97) Fecha y número de publicación de la concesión europea: **02.11.2022 EP 3799431**

(54) Título: **Preprocesamiento de datos de imágenes**

(30) Prioridad:

30.09.2019 US 201962908178 P

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
23.02.2023

(73) Titular/es:

ISIZE LIMITED (100.0%)
3 Falconet Court 123 Wapping High Street
London E1W 3NX, GB

(72) Inventor/es:

CHADHA, AARON y
ANDREOPOULOS, IOANNIS

(74) Agente/Representante:

PONS ARIÑO, Ángel

DESCRIPCIÓN

Preprocesamiento de datos de imágenes

Campo técnico

La presente divulgación se refiere a procedimientos implementados por ordenador para preprocesar datos de imágenes antes de su codificación con un codificador externo. La divulgación puede aplicarse en particular, pero no exclusivamente, cuando los datos de imágenes son datos de vídeo.

Antecedentes

Cuando se transmite de manera continua un conjunto de imágenes o vídeo a través del

protocolo HTTP de Internet, o a través de una conexión dedicada de conmutación de paquetes IP o de conmutación de circuitos usando otros protocolos, se debe seleccionar una serie de configuraciones de transmisión en flujo continuo y de codificación para garantizar el mejor uso posible del ancho de banda disponible. Para lograr esto, (i) el codificador de imágenes o vídeo debe ajustarse para proporcionar algún mecanismo de control de la velocidad de bits y *(ii)* el servidor de transmisión en flujo continuo debe proporcionar los medios para controlar o cambiar el flujo cuando el ancho de banda de la conexión no es suficiente para los datos transmitidos. Los procedimientos para abordar el control de la velocidad de bits en (i) incluyen [5]-[8]: codificación de velocidad de bits constante (CBR), codificación de velocidad de bits variable (VBR) o soluciones basadas en un modelo de verificador de memoria intermedia de vídeo (VBV) [5]-[8], tales como QVBR, CABR, CRF limitado, etc. Todas estas soluciones controlan los parámetros de la cuantificación adaptativa y la intrapredicción o interpredicción por imagen [5]-[8] con el fin de proporcionar la mejor precisión de reconstrucción posible para las imágenes o vídeos descodificados con el menor número de bits. Los procedimientos para abordar la adaptación de la transmisión en flujo continuo en *(ii)* son los protocolos DASH y HLS, es decir, para el caso de la transmisión adaptativa en flujo continuo a través de HTTP. En la transmisión adaptativa en flujo continuo, la adaptación incluye la selección de una serie de resoluciones de codificación, velocidades de bits y plantillas de codificación. Por lo tanto, el proceso de codificación y de transmisión en flujo continuo está destinado a cambiar el contenido de frecuencia del vídeo de entrada e introducir (de manera ideal) una pérdida de calidad imperceptible o (con suerte) controlable a cambio de un ahorro en la velocidad de bits. Esta pérdida de calidad se mide con una serie de métricas de calidad, que van desde métricas de relación de señal/ruido de bajo nivel hasta combinaciones complejas de métricas expertas que capturan elementos de mayor nivel de atención visual y percepción humanas. Una de esas métricas que es bien reconocida actualmente por la comunidad de vídeo y el Grupo de Expertos en Calidad de Vídeo (VQEG) es la Fusión de Evaluación Multimétodo de Vídeo (VMAF), propuesta por Netflix. Se ha trabajado mucho para convertir la VMAF en una métrica "autointerpretable": valores cercanos a 100 (por ejemplo, 93 o más) significan que el contenido comprimido es visualmente indistinguible del original, mientras que valores bajos (por ejemplo, por debajo de 70) significan que el contenido comprimido tiene una pérdida significativa de calidad en comparación con el original. Se ha notificado [Ozer, Streaming Media Mag., "Buyers' Guide to Video Quality Metrics", 29 de marzo de 2019] que una diferencia de alrededor de 6 puntos en VMAF corresponde a la denominada Mínima Diferencia Perceptible (JND), es decir, la diferencia de calidad que notará el espectador.

El proceso de codificación y descodificación a menudo requiere el uso de filtros lineales para la producción del contenido descodificado (y a menudo escalado de manera ascendente) que el espectador ve en su dispositivo. Desafortunadamente, esto tiende a provocar una fluctuación de calidad no controlada en la reproducción de vídeo o una reproducción de vídeo de mala calidad en general. Un espectador a menudo experimentará esto cuando lo vea en un dispositivo móvil, debido a una conexión y/o ancho de banda irregulares. Por ejemplo, moverse dentro o a través de un área con una intensidad de señal 4G/WiFi deficiente o similar puede hacer que una codificación de alta velocidad de bits de un flujo 4K conmute rápidamente a una codificación de velocidad de bits mucho más baja/resolución más baja, donde el descodificador y el reproductor de vídeo proseguirán con el escalado ascendente hasta la resolución del dispositivo de visualización mientras el espectador continúa con el visionado.

Las soluciones técnicas a este problema pueden agruparse en tres categorías distintas.

El primer tipo de enfoques consiste en soluciones que intentan una mejora basada en dispositivo, es decir, el avance del estado de la técnica en el escalado ascendente de vídeo inteligente en el reproductor de vídeo cuando el contenido se ha escalado de manera descendente "toscamente" usando un filtro lineal como el bicúbico o variantes de los Lanczos u otros filtros polifásicos [1]-[3],[10]-[15]. Estos enfoques incluyen soluciones SoC integradas en los últimos televisores 8K. Si bien ha habido algunos avances prometedores en este ámbito, este tipo de enfoque está limitado por las estrictas limitaciones de complejidad y consumo de energía de los componentes electrónicos de consumo. Además, dado que el contenido recibido en el cliente ya está distorsionado por la compresión (a menudo severamente), hay límites teóricos en relación con el nivel de detalle de las imágenes que se puede recuperar mediante el escalado

ascendente en el lado de cliente.

Una segunda familia de enfoques consiste en el desarrollo de codificadores de imágenes y vídeo personalizados, típicamente basados en redes neuronales profundas [10][12][1][13][14]. Esto se desvía de las normas de codificación, de empaquetado de flujos y de transporte de flujos y crea formatos personalizados, por lo que tiene la desventaja de requerir mecanismos de transporte personalizados y descodificadores personalizados en los dispositivos cliente. Además, en los más de 50 años de codificación de vídeo se han desarrollado muchas oportunidades para mejorar la ganancia en diferentes situaciones, lo que hace que el estado de la técnica actual en la predicción espacio-temporal y la codificación sean muy difíciles de superar con soluciones de redes neuronales que están diseñadas desde cero y aprenden de los datos.

La tercera familia de procedimientos comprende la optimización perceptiva de codificadores basados en normas existentes mediante el uso de métricas perceptivas durante la codificación. Aquí, los desafíos son los siguientes:

i) el ajuste requerido está considerablemente limitado por la necesidad de cumplir con la norma utilizada;
ii) muchas de las soluciones propuestas tienden a limitarse a modelos de focalización de la atención o a procedimientos de aprendizaje poco profundos con capacidad limitada, que suponen, por ejemplo, que la mirada humana se centra en áreas particulares del cuadro (por ejemplo, en un vídeo conversacional tendemos a mirar al/a los orador(es), no al fondo) o que usan algunos filtros definidos manualmente para mejorar los segmentos de imágenes o grupos de macrobloques de imágenes antes de su codificación; y
iii) dichos procedimientos tienden a requerir múltiples pasadas de codificación, aumentando así la complejidad.

Debido a estos problemas, los diseños conocidos están muy estrechamente ligados a la implementación específica del codificador. Rediseñarlos para un nuevo codificador y/o una nueva norma, por ejemplo la codificación de HEVC a VP9, puede requerir un esfuerzo considerable.

La presente divulgación busca resolver o mitigar algunos o todos de estos problemas mencionados anteriormente. De manera alternativa y/o adicional, los aspectos de la presente divulgación buscan proporcionar procedimientos mejorados para procesar datos de imágenes y, en particular, procedimientos que se puedan utilizar en combinación con los diseños de códec existentes de imágenes y vídeo.

El documento US 2012/0033040 A1, publicado el 9 de febrero de 2012, se refiere a procedimientos de selección de filtros y selectores de filtros para el preprocesamiento de vídeo en aplicaciones de vídeo. Una región de una imagen de entrada se preprocesa mediante múltiples filtros de preprocesamiento y la selección del filtro de preprocesamiento para una codificación posterior se basa en la métrica evaluada de la región.

El documento GB 2548749 A, publicado el 27 de septiembre de 2017, se refiere a la comunicación de datos visuales o de vídeo a través de una red. En un nodo receptor, se recibe una sección de menor calidad de datos visuales o de imágenes a través de una red, se recibe un algoritmo jerárquico correspondiente que puede hacerse funcionar para aumentar la calidad de los datos visuales de menor calidad, y se usa el algoritmo jerárquico para aumentar la calidad de los datos visuales de menor calidad recibidos escalando de manera ascendente la calidad o resolución espacial de las imágenes de menor calidad.

El documento US 2019/075301 A1, publicado el 7 de marzo de 2019, se refiere a una cadena de procesamiento de codificación de vídeo que incluye una cadena de procesamiento de codificación principal que comprime los datos de imágenes de origen correspondientes a un cuadro de imagen mediante el procesamiento de los datos de imágenes de origen en función de, al menos en parte, los parámetros de codificación para generar datos de imágenes codificados. Además, la cadena de procesamiento de codificación de vídeo incluye un bloque de aprendizaje automático acoplado comunicativamente a la cadena de procesamiento de codificación principal, en la que el bloque de aprendizaje automático analiza el contenido del cuadro de imagen procesando los datos de imágenes de origen en función de, al menos en parte, los parámetros de aprendizaje automático implementados en el bloque de aprendizaje automático cuando el bloque de aprendizaje automático está habilitado por los parámetros de codificación. La cadena de procesamiento de codificación de vídeo ajusta de forma adaptativa los parámetros de codificación en función de, al menos en parte, el contenido que se espera que esté presente en el cuadro de imagen para facilitar la mejora de la eficacia de codificación.

El documento de literatura que no es de patente XP011724474 (KIM MINHOE et AL: "Toward the Realization of Encoder and Decoder Using Deep Neural Networks", *IEEE COMMUNICATIONS MAGAZINE*, CENTRO DE SERVICIO DE IEEE, PISCATAWAY, EE.UU., vol. 57, n.º 5, 1 de mayo de 2019 (01/05/2019), páginas 57-63) se refiere a autocodificadores para comunicaciones en redes neuronales profundas (codificadores y descodificadores basados en DNN). Considera los componentes principales de las comunicaciones basadas en el aprendizaje profundo, hasta el diseño práctico de circuitos digitales, incluidos los resultados de la implementación y los experimentos.

Resumen

Los aspectos de la presente invención se definen en las reivindicaciones adjuntas. De acuerdo con un primer aspecto de la presente divulgación, se proporciona un procedimiento de preprocesamiento implementado por ordenador, antes de la codificación usando un codificador externo, de datos de imágenes usando una red de preprocesamiento que comprende un conjunto de pesos interconectados, comprendiendo el procedimiento:

recibir, en la red de preprocesamiento, datos de imágenes de una o más imágenes; y
procesar los datos de imágenes usando la red de preprocesamiento para generar una representación de píxeles de salida para su codificación con el codificador externo,
donde los pesos de la red de preprocesamiento

están entrenados para optimizar una combinación de:
al menos una puntuación de calidad indicativa de la calidad de la representación de píxeles de salida; y una puntuación de velocidad indicativa de los bits requeridos por el codificador externo para codificar la representación de píxeles de salida.

Mediante el uso de una red de preprocesamiento de la manera descrita, se puede mejorar la calidad visual de los datos de imágenes codificados y descodificados posteriormente para una velocidad de bits de codificación dada, y/o se puede reducir una velocidad de bits de codificación para lograr una calidad visual dada. En particular, se puede mejorar la calidad objetiva, perceptiva y/o estética de imágenes o vídeos descodificados posteriormente. La fidelidad de los datos de imágenes codificados y descodificados posteriormente con respecto a los datos de imágenes originales también se puede mejorar mediante el uso de los procedimientos descritos en el presente documento. El uso de pesos entrenados para optimizar las puntuaciones de calidad y de velocidad mejora el rendimiento de la red de preprocesamiento y permite que el preprocesamiento de los datos de imágenes se realice de manera óptima para hacer que el codificador externo (que puede ser un codificador basado en normas) funcione de la manera más eficiente posible. Además, la representación de píxeles de salida se puede escalar de manera ascendente mediante un dispositivo cliente usando sus filtros lineales existentes.

Los procedimientos descritos incluyen soluciones técnicas que pueden aprender en función de los datos y pueden utilizar un codificador de imágenes/vídeo estándar con una configuración de codificación predeterminada. Una cuestión técnica general abordada puede resumirse así: ¿cómo preprocesar (o "precodificar") de manera óptima el flujo de píxeles de un vídeo en un flujo de píxeles (típicamente) más pequeño con el fin de hacer que los codificadores basados en normas sean lo más eficientes (y rápidos) posible? Esta pregunta puede ser especialmente relevante cuando el dispositivo cliente puede escalar de manera ascendente el contenido con sus filtros lineales existentes, y/o cuando la calidad perceptiva se mide con los últimos avances en las métricas de calidad perceptiva de la literatura, por ejemplo, usando VMAF o métricas similares.

De manera ventajosa, la al menos una puntuación de calidad indica la distorsión de señal en la representación de píxeles de salida. Por ejemplo, los datos de imágenes pueden preprocesarse de modo que se minimice la distorsión de la señal en la representación de los píxeles de salida. En realizaciones, la al menos una puntuación de calidad indica la pérdida de la calidad estética o perceptiva en la representación de píxeles de salida.

En realizaciones, la resolución de la representación de píxeles de salida aumenta o disminuye de acuerdo con una relación de escalado ascendente o descendente. Por ejemplo, la resolución de la representación de píxeles de salida puede ser menor que la resolución de los datos de imágenes recibidos. Al escalar de manera descendente la imagen antes de usar el codificador externo, el codificador externo puede funcionar de manera más eficiente procesando una imagen de menor resolución. Por otra parte, los parámetros usados durante el escalado descendente/ascendente se pueden elegir para proporcionar diferentes resultados deseados, por ejemplo para mejorar la precisión (es decir, el grado de similitud entre las imágenes recuperadas y las originales). Además, el proceso de escalado descendente/ascendente puede diseñarse para que esté en conformidad con el escalado descendente/ascendente realizado por el codificador externo, de modo que el codificador externo pueda codificar las imágenes escaladas de manera descendente/ascendente sin que se pierda información esencial.

En realizaciones, la representación de píxeles de salida se corrompe mediante la aplicación de una o más funciones matemáticamente diferenciables y una aproximación, en donde la representación de píxeles de salida se corrompe para aproximarse a la corrupción esperada de una transformada basada en bloques y una cuantificación usada en el codificador externo, y/o para aproximarse a la corrupción esperada de una transformada y una cuantificación de errores calculados a partir de un proceso de predicción temporal basado en bloques usado en el codificador externo. La corrupción de la representación de píxeles de salida introduce una pérdida de fidelidad (por ejemplo, el bloqueo de artefactos) que emula las pérdidas de fidelidad introducidas a partir de codificadores típicos usados para comprimir

datos de imágenes o vídeo. Esto permite que el sistema divulgado utilice dicho comportamiento emulado en su proceso de funcionamiento y optimización.

En realizaciones, la representación de píxeles de salida se redimensiona a la resolución de los datos de imágenes de entrada, usando un filtro lineal o no lineal configurado durante una fase inicial de configuración o entrenamiento. Esto puede permitir cuantificar las puntuaciones de calidad para la representación de píxeles de salida (es decir, evaluando la calidad de la salida redimensionada). El redimensionamiento puede emular un proceso en un descodificador cliente para mostrar el contenido de píxeles a una resolución dada.

De manera ventajosa, la red de preprocesamiento comprende una red neuronal artificial que incluye múltiples capas que tienen una arquitectura convolucional, donde cada capa está configurada para recibir la salida de una o más capas anteriores. En realizaciones, las salidas de cada capa de la red de preprocesamiento se hacen pasar a través de una función rectificadora lineal paramétrica no lineal, pReLU. En otras realizaciones pueden usarse otras funciones no lineales.

En realizaciones, durante una fase de configuración o entrenamiento inicial: la al menos una puntuación de calidad se optimiza en una dirección de reconstrucción o calidad visual mejoradas; y la puntuación de velocidad se optimiza en una dirección de velocidad más baja. En realizaciones, una de la al menos una puntuación de calidad y la puntuación de velocidad se fija durante el entrenamiento, y la otra de la al menos una puntuación de calidad y la puntuación de velocidad se optimiza. En otras realizaciones, se optimizan tanto la al menos una puntuación de calidad como la puntuación de velocidad.

En realizaciones, la al menos una puntuación de calidad y la puntuación de velocidad se optimizan de acuerdo con un procedimiento de optimización lineal o no lineal que ajusta los pesos de la red de preprocesamiento y/o ajusta el tipo de la arquitectura usada para interconectar los pesos de la red de preprocesamiento.

En realizaciones, la representación de píxeles de salida se codifica con el codificador externo. En realizaciones, la representación de píxeles codificados se proporciona para su transmisión, por ejemplo a un descodificador, para su posterior descodificación y visualización de los datos de imágenes. En realizaciones, el codificador externo es un codificador estándar ISO JPEG o ISO MPEG, o un codificador AOMedia.

En realizaciones, la representación de píxeles de salida se filtra usando un filtro lineal, donde el filtro lineal comprende un filtro de desenfoque o de mejora de bordes. Dicho filtro puede emular un filtro aplicado en un descodificador y/o dispositivo de visualización.

En realizaciones, la al menos una puntuación de calidad incluye uno o más de lo siguiente: relación máxima de señal/ruido, métrica de índice de similitud estructural (SSIM), métricas de calidad multiescala, métrica de pérdida de detalle o SSIM multiescala, métricas basadas en múltiples puntuaciones de calidad y aprendizaje y entrenamiento basados en datos, fusión de evaluación multimétodo de vídeo (VMAF), métricas de calidad estética.

En realizaciones, la al menos una puntuación de calidad y la puntuación de velocidad se combinan con pesos lineales o no lineales, en donde los pesos lineales o no lineales se entrenan en función de procedimientos de retropropagación y de descenso de gradiente con datos de entrenamiento representativos. Esto permite optimizar la configuración y/o el funcionamiento de la red de preprocesamiento.

De acuerdo con otro aspecto de la divulgación, se proporciona un procedimiento implementado por ordenador que transforma grupos de datos de entrada de píxeles de una única o una pluralidad de cuadros de imágenes o vídeo en otros grupos de datos de píxeles, donde los datos de salida transformados se optimizan en ambos de los siguientes conjuntos de puntuaciones:

(i) un intervalo de puntuaciones que representa la distorsión de señal, la calidad perceptiva o estética, ya sea de manera independiente (es decir, sin usar cuadros de imágenes o vídeo de referencia con los que comparar), o en función de cuadros de imágenes o vídeo de referencia;
(ii) una o más puntuaciones que representan los bits necesarios para codificar, es decir, comprimir, los grupos de salida transformados de píxeles basados en un codificador externo de imágenes o vídeo; esta velocidad de bits se mide habitualmente en bits por píxel (bpp) o en bits por segundo (bps), si los cuadros de imágenes o vídeo se procesan/codifican o se muestran con una determinada velocidad en el tiempo (como es habitual en las señales de vídeo).

En realizaciones, el procedimiento de transformación divulgado es un procedimiento implementado por ordenador, que comprende:

(i) una *etapa de redimensionamiento opcional,* que puede ser un proceso de escalado ascendente o descendente, o un redimensionamiento con una velocidad igual a la unidad para evitar el redimensionamiento;
(ii) un *mapeo de píxel a píxel* basado en combinaciones lineales o no lineales de pesos, que están interconectados en una red y pueden incluir no linealidades tales como funciones de activación y capas de agrupación, lo que hace que este mapeo corresponda a un diseño de red neuronal artificial.

Con respecto a la etapa de redimensionamiento opcional, el redimensionamiento puede tener lugar con cualquier número entero o fraccionario y usa un conjunto de pesos de filtro para filtrar y redimensionar la imagen, que puede ser un filtro lineal o un filtro no lineal, tal como una red neuronal artificial que se ha entrenado con datos con el fin de lograr un escalado descendente o ascendente de los grupos de píxeles de entrada de acuerdo con un criterio de minimización de distorsión, tal como minimizar el error cuadrático medio y el error de gradientes de borde o píxeles en las imágenes o vídeos de acuerdo con las señales de entrenamiento proporcionadas y el entrenamiento fuera de línea.

La salida transformada se puede corromper opcionalmente, con el fin de introducir la pérdida de fidelidad, de una manera que emula la introducida por un codificador típico de imágenes o vídeo en uso con sistemas comerciales de compresión de imagen/vídeo. Ejemplos de esta pérdida de fidelidad son, pero no se limitan a, el bloqueo de artefactos de compensación de movimiento en vídeo, desenfoque y cambios de frecuencia en los datos de píxeles de entrada debido a la transformación y cuantificación realizada en codificadores típicos de JPEG, MPEG o AOMedia, y otra pérdida de fidelidad geométrica y de textura debido a procedimientos de predicción intracuadro usados en los codificadores mencionados anteriormente. Esta pérdida de fidelidad se puede diseñar para emular las pérdidas de codificación esperadas de codificadores típicos usados para comprimir datos de imágenes o vídeo en un intervalo de velocidades de bits de codificación, y permite que la invención y el sistema divulgados representen dicho comportamiento emulado en su proceso de diseño y optimización como un programa informático que implementa estas funciones.

Con el fin de hacer que el proceso de optimización pueda aprender en función de los datos y el entrenamiento fuera de línea con datos almacenados, o el entrenamiento periódico con datos recién adquiridos, las funciones que calculan las puntuaciones de calidad y velocidad y las funciones que emulan la pérdida de fidelidad se pueden hacer matemáticamente diferenciables, de modo que se pueden usar técnicas que actualizan los pesos en una red neuronal artificial basada en la diferenciación de salida y la propagación de errores [10],[12]. Si algunas funciones que emulan la pérdida de fidelidad no son matemáticamente diferenciables, donde un ejemplo típico es un cuantificador de bloques de múltiples etapas, se pueden convertir en funciones matemáticamente diferenciables a través de aproximaciones diferenciables continuas.

Con el fin de cuantificar las puntuaciones de calidad para los píxeles transformados (y opcionalmente corruptos) de la invención y el sistema proporcionados, la salida se puede redimensionar a las dimensiones originales de cuadro de imagen o vídeo usando un procedimiento de redimensionamiento lineal o no lineal, tal como un filtro o una red neuronal artificial, y las puntuaciones de calidad se evalúan en la salida redimensionada. Sin embargo, las puntuaciones de velocidad se evalúan preferentemente en la salida del mapeo de píxel a píxel, antes de cualquier redimensionamiento a las dimensiones originales de cuadro de imagen o vídeo. Esto se debe a que la puntuación de velocidad y la estimación de velocidad se refieren a la codificación de la salida transformada por un codificador estándar de imágenes o vídeo (antes de la transmisión a través de la red), mientras que el redimensionamiento emula un proceso en el descodificador cliente para mostrar el contenido de píxeles a la resolución requerida.

Con el fin de optimizar los pesos y parámetros utilizados asociados a la puntuación de calidad y velocidad, y cualquier parámetro asociado a la emulación de pérdida de fidelidad, las entradas representativas se pueden procesar de manera iterativa bajo una función objetivo dada en el intervalo de puntuaciones de calidad y mediciones de la velocidad de bits de codificación. Esto se puede hacer fuera de línea en una fase de configuración o entrenamiento, pero también se puede repetir en línea varias veces durante el funcionamiento del sistema, con el fin de ajustarse al contenido o a los dispositivos de codificación específicos, o ajustar de forma precisa los pesos de transformación ya establecidos y utilizados y las funciones de puntuación de calidad-velocidad y las funciones de pérdida de fidelidad. Este entrenamiento se puede realizar usando procedimientos de regresión estadística o de ajuste. Es importante destacar que también se puede realizar usando cualquier combinación de aprendizaje por retropropagación y actualizaciones de descenso de gradiente de pesos o errores calculados a partir de las puntuaciones utilizadas y las funciones de pérdida de calidad. El aprendizaje por retropropagación puede usar reglas de aprendizaje que son deterministas o estocásticas (aleatorias) y los gradientes se pueden calcular con entradas individuales, con lotes de entradas o con todo el conjunto de datos de entrenamiento, por cada iteración de entrenamiento. Los parámetros de aprendizaje, tales como la velocidad de aprendizaje inicial y la disminución de la velocidad de aprendizaje, se pueden ajustar empíricamente para optimizar la velocidad del entrenamiento y el rendimiento. Los lotes de datos de entrenamiento se pueden seleccionar de manera determinista o aleatoria/pseudoaleatoria.

En lo que respecta a puntuaciones de calidad reales que pueden ser usadas por los procedimientos y el sistema

divulgados, estas incluyen, pero no se limitan a, una o más de las siguientes puntuaciones de calidad de imagen objetiva, perceptiva o estética: relación máxima de señal/ruido (PSNR), métrica de índice de similitud estructural (SSIM), métricas de calidad multiescala, tales como la métrica de pérdida de detalle o la SSIM multiescala, métricas basadas en múltiples puntuaciones de calidad y en aprendizaje y entrenamiento basados en datos, tales como la fusión de evaluación multimétodo de vídeo (VMAF), o métricas de calidad estética [13], y variaciones de estas métricas. Las puntuaciones de calidad pueden estar basadas, o no, en referencias, y cada puntuación de calidad se optimiza en la dirección correspondiente al aumento de la calidad visual.

En lo que respecta a puntuaciones de velocidad, incluyen, pero no se limitan a, estimaciones de la velocidad de bpp para codificar la nueva representación de píxeles producida por la invención divulgada con un conjunto de modelos que usan funciones logarítmicas, armónicas o exponenciales para modelar los bpp o bps esperados de un codificador estándar de imágenes o vídeo, pero también mezclas de dichas puntuaciones con modelos operativos que emulan la codificación entrópica utilizada por dichos codificadores, donde algunos ejemplos son la emulación de codificación aritmética adaptada al contexto, la codificación Huffman, la longitud de ejecución y la codificación predictiva. Los modelos analíticos y/u operativos que expresan o emulan la velocidad esperada para codificar las salidas transformadas de la invención se pueden convertir en funciones matemáticamente diferenciables, que se pueden entrenar con procedimientos de retropropagación y descenso de gradiente y datos de entrenamiento que son representativos de la velocidad de bpp o bps del codificador utilizado para comprimir la representación de píxeles transformados producidos por la invención divulgada. La puntuación de velocidad se puede optimizar minimizando la velocidad de bbp o bps, lo que se puede hacer de una de tres maneras: *(i)* minimizando directamente la puntuación de velocidad sin restricciones; *(ii)* minimizando la puntuación de velocidad sujeta a una restricción de velocidad fija global sobre todos los datos de entrada; *(iii)* minimizando la distancia entre la puntuación de velocidad y una puntuación de velocidad de referencia por cada imagen o vídeo de entrada.

Las múltiples puntuaciones de calidad y de velocidad se pueden combinar en una sola función objetivo usando funciones tanto lineales como no lineales. Ejemplos de funciones lineales son las sumas de las puntuaciones con coeficientes de pesos. Ejemplos de funciones no lineales son las sumas de funciones no lineales de estas puntuaciones usando funciones logarítmicas, exponenciales, sigmoides, armónicas u otros tipos de funciones no lineales. Los parámetros de los pesos y funciones que combinan las múltiples puntuaciones de calidad y velocidad también pueden ser parámetros entrenables que se pueden optimizar usando procedimientos de retropropagación y de descenso de gradiente, con datos de entrenamiento representativos correspondientes a resultados de calidad y velocidad de las métricas de calidad y los modelos de velocidad. Más allá de dicho entrenamiento, los parámetros también se pueden optimizar usando regresión lineal o no lineal, pruebas de ajuste estadístico y procedimientos bayesianos que incorporan conocimiento previo acerca de los datos de entrada en el modelo.

En realizaciones, el codificador externo comprende un códec de imágenes. En realizaciones, los datos de imágenes comprenden datos de vídeo y las una o más imágenes comprenden cuadros de vídeo. En realizaciones, el codificador externo comprende un códec de vídeo.

Los procedimientos de procesamiento de datos de imágenes descritos en el presente documento se pueden realizar en un lote de datos de vídeo, por ejemplo, un archivo de vídeo completo para una película o similar, o en un flujo de datos de vídeo.

De acuerdo con otro aspecto de la divulgación, se proporciona un dispositivo informático, que comprende:

un procesador; y
memoria;
en donde el dispositivo informático está dispuesto para funcionar usando el procesador cualquiera de los procedimientos descritos anteriormente.

De acuerdo con otro aspecto de la divulgación, se proporciona un producto de programa informático dispuesto para, cuando se ejecuta en un dispositivo informático que comprende un proceso o memoria, realizar cualquiera de los procedimientos descritos anteriormente.

Por supuesto, se apreciará que las características descritas en relación con un aspecto de la presente divulgación descrito anteriormente se pueden incorporar en otros aspectos de la presente divulgación.

Descripción de los dibujos

A continuación se describirán realizaciones de la presente divulgación a modo de ejemplo únicamente con referencia a los dibujos esquemáticos adjuntos, de los cuales:

La figura 1 es un diagrama esquemático de un procedimiento de procesamiento de datos de imágenes de acuerdo con realizaciones;
Las figuras 2(a) a 2(c) son diagramas esquemáticos que muestran una red de preprocesamiento de acuerdo con realizaciones;
La figura 3 es un diagrama esquemático que muestra una red de preprocesamiento según las realizaciones; las figuras 4 a 6 son diagramas esquemáticos que muestran ejemplos de procesos de entrenamiento según las realizaciones;
Las figuras 7 a 9 son gráficos de resultados de calidad frente a velocidad de bits de acuerdo con realizaciones;
La figura 10 es un diagrama de flujo que muestra las etapas de un procedimiento de preprocesamiento de datos de imágenes de acuerdo con realizaciones; y
La figura 11 es un diagrama esquemático de un dispositivo informático de acuerdo con realizaciones.

Descripción detallada

A continuación se describen realizaciones de la presente divulgación.

La figura 1 es un diagrama esquemático que muestra un procedimiento de procesamiento de datos de imágenes, de acuerdo con realizaciones. Los datos de entrada de imagen o vídeo se codifican y descodifican con un codificador externo de imágenes o vídeo. Las realizaciones ilustradas pueden aplicarse al procesamiento por lotes, es decir, al procesamiento conjunto de un grupo de cuadros de imágenes o vídeo sin restricciones de retardo (por ejemplo, una secuencia de vídeo completa), así como al procesamiento de flujos, es decir, al procesamiento de solo un subconjunto limitado de un flujo de cuadros de imágenes o vídeo, o incluso un subconjunto seleccionado de una única imagen, debido a restricciones de retardo o almacenamiento en memoria intermedia. El procedimiento representado en la figura 1 incluye precodificación profunda, antes de la codificación con el codificador externo, con optimización de la puntuación de calidad-velocidad (y redimensionamiento opcional) en la cadena de procesamiento de transmisión.

El primer componente que procesa los cuadros de imagen o de vídeo de entrada comprende la precodificación profunda con pérdida de calidad-velocidad (también denominada "pérdida Q-R", como se representa en la figura 1). Esta precodificación consiste en un redimensionador y un componente optimizador de calidad-velocidad profunda. El primero puede escalar de manera descendente o ascendente la entrada usando un filtro no lineal, o una red neuronal artificial basada en los parámetros *s* proporcionados. Si $s < 1$, el redimensionador escala de manera ascendente los bloques de píxeles de entrada en 1/s, por ejemplo si s = 0,25, cada píxel de entrada se convertirá en 4 píxeles a la salida del redimensionador. Si $s > 1$, entonces el redimensionador escala de manera descendente en s, es decir, en promedio, los píxeles s se convertirán en 1 píxel después del redimensionador. El valor de s puede proporcionarse externamente o puede ajustarse en otras realizaciones, y s puede ser cualquier número fraccionario, pero también puede ser la unidad (s = 1), donde esto último no implica redimensionamiento alguno. El efecto del redimensionador se invierte en el componente de redimensionamiento posterior a la descodificación que se muestra en el lado derecho de la figura 1 y los grupos de píxeles recuperados pueden formar una imagen recuperada de la resolución original que se mostrará a un espectador después de un componente de procesamiento posterior opcional, que puede ser un filtro lineal o no lineal o una red neuronal artificial que mejora los aspectos estéticos o perceptivos de la imagen recuperada.

Entre la salida de la precodificación de vídeo profunda con pérdida Q-R de la figura 1 y el descodificador, se usa un codificador externo de imágenes o vídeo, que puede comprender cualquier codificador ISO JPEG o ISO MPEG o AOMedia, o cualquier otro codificador propietario. Además, como se muestra en la figura 1, el flujo de bits producido desde el codificador puede almacenarse o transmitirse a través de una red al descodificador correspondiente.

El optimizador profundo de calidad-velocidad (DQRO) mostrado en la figura 1 puede comprender cualquier combinación de pesos conectados en una red y que tiene una función no lineal (similar a una función de activación de una red neuronal artificial). En la figura 2(a) se muestra un ejemplo de dichos pesos. El DQRO entrenado comprende múltiples capas de pesos y funciones de activación. En la figura 2(b) se muestra un ejemplo de la conectividad entre pesos y entradas. Es decir, la figura 2(a) muestra una combinación de entradas $x_0, \ldots, x_3$ con coeficientes de peso Θ y una función de activación no lineal g(), y la figura 2(b) es un diagrama esquemático que muestra capas de activaciones y pesos interconectados que forman una red neuronal artificial. Dichos ejemplos se entrenan con retropropagación de errores calculados en la capa de salida, usando procedimientos de descenso de gradiente. Esto se muestra en la figura 2(c), que representa esquemáticamente la retropropagación de errores δ desde una capa intermedia (lado derecho de la figura 2(c)) hasta la capa intermedia anterior usando el descenso de gradiente.

En la figura 3 se muestra un ejemplo de la precodificación condicional profunda. Consiste en una cascada de capas convolucionales (Conv (*k* 3 k)) y de ReLu paramétricas (pReLu) de pesos y funciones de activación que mapean grupos de píxeles de entrada a grupos de píxeles de salida transformados, similar al ejemplo mostrado en la figura 2(b). Las capas convolucionales extienden el ejemplo de la figura 2(b) a múltiples dimensiones, realizando operaciones de convolución entre filtros multidimensionales de tamaño de núcleo fijo (*k* 3 *k*) con pesos que se pueden aprender y

las entradas a la capa. Cada activación en la salida de la capa convolucional solo tiene conectividad local (no global) a una región local de la entrada. La conectividad de la cascada de capas convolucionales y las funciones de activación también puede incluir conexiones de salto, como se muestra mediante la conexión desde la salida de la capa "Conv (3 3 3)" más a la izquierda de la figura 3 hasta el punto de suma de la figura 3. Además, la totalidad de la cascada de múltiples capas (también conocida como red neuronal profunda) se puede entrenar de extremo a extremo en función de la retropropagación de errores desde la capa de salida hacia atrás (por ejemplo, como se muestra en la figura 2(c)), usando procedimientos de descenso de gradiente.

Las figuras 4 y 5 representan procedimientos de entrenamiento de la red de preprocesamiento (es decir, el modelo de precodificación profunda) de acuerdo con realizaciones. En la figura 4 no se usa predicción temporal, mientras que en la figura 5 se usa predicción temporal para intercuadros en secuencias de vídeo. La mitad superior de cada una de las figuras 4 y 5 muestra el entrenamiento del modelo de precodificación profunda con el modelo perceptivo utilizado sin entrenar, es decir, en un estado "congelado". La mitad inferior de cada una de las figuras 4 y 5 ilustra el entrenamiento del modelo perceptivo, con el modelo de precodificación profunda sin entrenar, es decir, en un estado congelado. Las flechas que se extienden entre la mitad superior y la mitad inferior de cada una de las figuras 4 y 5 representan iteraciones de actualizaciones de peso entre el entrenamiento del modelo perceptivo y el modelo de precodificación. El proceso de entrenamiento global consiste en el entrelazamiento entre el entrenamiento de un modelo y la congelación del otro, y el refinamiento iterativo de ambos modelos a través de este proceso. El sistema global de entrenamiento tiene múltiples componentes, que se analizarán a continuación de manera separada.

El modelo perceptivo comprende dos partes; ambas partes toman como entrada la imagen de entrada, x, y una imagen distorsionada y optimizada por DQRO, $\hat{x}$, y estiman una serie de puntuaciones objetivas, subjetivas o estéticas para la imagen $\hat{x}$.

Las puntuaciones pueden ser puntuaciones basadas en referencias, es decir, puntuaciones que comparan x con x, pero también pueden ser puntuaciones sin referencias, como se emplea en los procedimientos de evaluación ciega de calidad de imagen. El modelo perceptivo puede aproximar funciones de puntuación perceptiva no diferenciables, incluidas VIF, ADM2 y VMAF, con funciones diferenciables continuas. El modelo perceptivo también se puede entrenar para proporcionar puntuaciones de evaluadores humanos, incluidas MOS o distribuciones a través de valores de ACR. Específicamente, el modelo perceptivo usa redes neuronales artificiales con pesos y funciones de activación y conectividad entre capas (por ejemplo, como se muestra en la figura 2(b)), pero también comprende extensiones o una disposición de múltiples módulos de este tipo, interconectados de manera paralela y secuencial (en cascada). Con el fin de entrenar el modelo perceptivo (mitad inferior de la figura 4), la pérdida perceptiva se minimiza L_P, que es la diferencia agregada (o error) entre las puntuaciones perceptivas vectorizadas predichas y las puntuaciones vectorizadas de referencia por cada entrada (a partir de cálculo numérico o evaluadores humanos). La función de pérdida entre las puntuaciones predichas y las de referencia puede estar basada en normas (por ejemplo, error cuadrático medio o error absoluto medio) o basada en distribución (por ejemplo, empleando un entrenamiento confrontativo con un discriminador para alinear las distribuciones predichas y de referencia a través del espacio métrico). Sin embargo, otras realizaciones de esta función de pérdida comprenden combinaciones no lineales de puntuaciones perceptivas usando funciones logarítmicas, armónicas, exponenciales y otras funciones no lineales. Con el fin de entrenar el modelo de precodificación profunda (mitad superior de la figura 4), las puntuaciones perceptivas predichas se combinan primero con las puntuaciones de fidelidad predichas que representan la reconstrucción estructural o en píxeles de la entrada x. Las puntuaciones de fidelidad, tales como SSIM, MS-SSIM y PSNR, son totalmente diferenciables y se pueden calcular directamente a partir de x y $\hat{x}$. El modelo de precodificación profunda (que incluye DQRO y redimensionamiento opcional) se entrena optimizando la pérdida de distorsión L_D con las puntuaciones perceptivas y de fidelidad ponderadas y combinadas. Específicamente, cada puntuación se maximiza o minimiza en la dirección de aumentar la calidad perceptiva o estética con el fin de lograr un equilibrio en $\hat{x}$ entre la mejora perceptiva sobre x y la reconstrucción fiel de x. La ponderación y combinación de puntuaciones en la figura 4 comprenden una función lineal del tipo c_1s_1, + c_2s_2 + ■■■ + c_Ns_N, donde $c_1, \ldots, c_N$ son los pesos y $s_1,\ldots, s_N$ son las puntuaciones de calidad predichas, y se aplican los mismos pesos para las puntuaciones medidas de la imagen de entrenamiento. Sin embargo, otros ejemplos de esta función de pérdida comprenden combinaciones no lineales de estas puntuaciones usando funciones logarítmicas, armónicas, exponenciales y otras funciones no lineales.

El modelo de precodificación profunda mostrado en el proceso de entrenamiento de las figuras 4 y 5 corresponde al diseño mostrado en la figura 3, que comprende un redimensionamiento opcional y un optimizador profundo de calidad-velocidad (DQRO), y corresponde al bloque de precodificación de vídeo profundo desplegado en la figura 1. Sin embargo, también son posibles otras variaciones. El entrenamiento del DQRO se lleva a cabo con retropropagación y cualquier variación del descenso de gradiente de la pérdida de distorsión ponderada y la pérdida de velocidad de la figura 4. Se aplican parámetros del proceso de aprendizaje, tales como la velocidad de aprendizaje, el uso de interrupciones y otras opciones de regularización para estabilizar el proceso de entrenamiento y convergencia.

También se usa un módulo de códec virtual en el diseño representado en las figuras 4 y 5. Dos ejemplos de este

módulo se ilustran en las figuras 4 y 5, respectivamente. El módulo de códec virtual de la figura 4 consiste en un componente de transformada de frecuencia, un componente de cuantificación y de codificación entrópica y un componente de descuantificación y de transformada inversa. El propósito del módulo de códec virtual es emular un codificador típico de imágenes o vídeo usando componentes diferenciables y con capacidad de aprendizaje, tales como las capas de una red neuronal artificial. El componente de transformada de frecuencia es cualquier variación de transformada discreta de seno o coseno o transformada de ondícula, o una descomposición basada en átomos. La descuantificación y el componente de transformada inversa pueden convertir los coeficientes de transformada en valores de píxeles aproximados. La principal fuente de pérdida para el módulo de códec virtual proviene del componente de cuantificación, que emula cualquier cuantificador de zona muerta o no de zona muerta de múltiples etapas. Finalmente, el componente de codificación entrópica puede ser una aproximación diferenciable continua de la entropía teórica (ideal) en relación con valores de transformada, o una representación diferenciable continua de un codificador Huffman, un codificador aritmético, un codificador de longitud de ejecución o cualquier combinación de aquellas que también se hace que sea adaptable al contexto, es decir, teniendo en cuenta los tipos de símbolos de cuantificación y los valores circundantes (condicionamiento del contexto) con el fin de usar el modelo de probabilidad y el procedimiento de compresión apropiados. La pérdida de velocidad L_R se calcula minimizando la velocidad predicha a partir del procesamiento del modelo de códec virtual (es decir, codificación y descodificación virtual) de los coeficientes cuantificados derivados de los píxeles DQRO, sujetos o no sujetos a una restricción de velocidad en el límite de velocidad superior. Esta pérdida de velocidad se optimiza en función de los pesos de precodificación profunda, mediante retropropagación usando variaciones de los procedimientos de descenso de gradiente, con el fin de entrenar la precodificación profunda. Más allá de su utilidad como estimador de velocidad, el módulo de códec virtual produce las salidas DQRO distorsionadas (o corruptas), es decir, la señal $\hat{x}$ en la figura 4, que se usa para entrenar la parte no diferenciable del modelo perceptivo (parte inferior de la figura 4). Además, antes de o durante el entrenamiento del propio DQRO, cualquier parámetro asociado al módulo de códec virtual también se puede ajustar empíricamente o entrenar con procedimientos de retropropagación y descenso de gradiente. Esto implica entrenar cualquier parámetro de transformada y cuantificación que sea diferenciable, y también los parámetros de red neuronal artificial usados para representar las operaciones matemáticas no diferenciables de las partes de transformada y cuantificación con aproximaciones diferenciables, mediante el uso de la velocidad real para codificar los mismos píxeles con un codificador abierto JPEG, MPEG o AOMedia con pérdidas como referencia.

El módulo de códec virtual de la figura 5 extiende el de la figura 4 mediante la incorporación de un módulo de predicción temporal antes de la transformada de frecuencia para emular la codificación de vídeo de intercuadros en una secuencia de vídeo. Específicamente, el módulo de predicción temporal recibe los píxeles de salida de DQRO y un cuadro de referencia. La diferencia entre el cuadro de referencia y la salida de DQRO se calcula por bloques y el cuadro de error se pasa a la transformada de frecuencia. La descuantificación y la transformada de frecuencia inversa también reciben el cuadro de referencia con el fin de reconstruir una representación de cuadros de la entrada x para el modelado perceptivo. En lo que respecta a los intracuadros, el módulo de predicción temporal simplemente puede tratarse como una función de identidad y omitirse, como en la figura 4.

En las realizaciones mostradas en las figuras 4 y 5, el modelo perceptivo y la precodificación profunda se entrenan en intervalos y, después de entrenar uno, sus pesos y parámetros actualizados se congelan mientras se entrena el otro. Esta actualización de peso y entrenamiento intercalado mejoran y permiten un entrenamiento de extremo a extremo y una mejora iterativa tanto durante la fase de entrenamiento como en cualquier momento durante el funcionamiento del sistema. Un ejemplo de esto es cuando se añaden nuevas imágenes y puntuaciones de calidad en el sistema o se añaden nuevas formas de transformada y modos de codificación entrópica y de cuantificador, que corresponden a una forma nueva o actualizada de codificación de imágenes o vídeo, o nuevos tipos de contenido de imagen, por ejemplo imágenes de dibujos animados, imágenes de juegos de ordenador, aplicaciones de realidad virtual o aumentada, etc. De forma alternativa, en lugar de un entrenamiento iterativo, el modelo perceptivo se puede entrenar previamente en ejemplos representativos y seguir congelado durante todo el entrenamiento de la precodificación profunda (es decir, solo la mitad superior de las figuras 4 y 5).

La figura 6 muestra una variante del ejemplo de la figura 4 para la superresolución de imágenes o vídeo perceptivamente mejorada y restringida por velocidad. En este caso, el objetivo es escalar de manera ascendente y óptima una imagen de baja resolución dada o un cuadro de vídeo x_{LR} y un factor de escala $s < 1$. Durante el entrenamiento, se proporcionan pares de entrada de baja resolución y alta resolución, x_{LR} y x_{HR}, donde el mapeo de funciones puede ser conocido o desconocido. Las entradas de baja resolución x_{LR} se escalan de manera ascendente mediante el redimensionador, que puede ser cualquier modelo de superresolución preentrenado existente u otra red neuronal artificial que comprenda múltiples capas convolucionales y funciones de activación. A continuación, la salida se pasa a través de un optimizador profundo de calidad-velocidad (DQRO) que representa un mapeo de píxel a píxel. La velocidad de la salida de DQRO se modela por tanto mediante un códec virtual y se optimiza con la pérdida de velocidad L_R, con el códec virtual descrito para la ilustración de la figura 4. La salida del códec virtual $\hat{x}_{HR}$ es una representación escalada de manera ascendente de la entrada de alta resolución x_{HR}. De este modo, x_{HR} y x_{HR} pueden pasarse al modelo perceptivo, y la calidad perceptiva y de reconstrucción puede optimizarse con pérdida de distorsión

L_D y entrenamiento iterativo con L_D y L_p, como se describió anteriormente con referencia a la figura 4. Por lo tanto, el módulo de precodificación profunda puede implementarse para la precodificación de vídeo profunda en un servidor remoto, por ejemplo, como se muestra en la figura 1, o como un reemplazo para el redimensionamiento posterior a la descodificación en el cliente, con el fin de generar imágenes o cuadros escalados de manera ascendente y perceptivamente mejorados.

Los resultados de realizaciones de ejemplo de la presente divulgación de invención incluyen, pero no se limitan a, aquellos presentados de la figura 7 a la figura 9, que utilizan el códec de vídeo MPEG/ITU-T H.264MEVC representado en la biblioteca FFmpeg libx265 de código abierto. Las curvas de calidad-velocidad de bits mostradas en las figuras 7-9 muestran ejemplos de resultados de calidad promedio frente a velocidad de bits logrados con los procedimientos divulgados en 12 secuencias de vídeo Full HD (1920x1080 píxeles) y sin redimensionamiento. Los resultados mostrados en la figura 7 usan como medida de calidad la métrica ADM2 de la biblioteca VMAF de Netflix. En la figura 8, la calidad se mide usando la métrica VIF de la biblioteca VMAF de Netflix. En la figura 9, la calidad se mide usando la métrica VMAF de la biblioteca VMAF de Netflix. En cada caso, la calidad se mide para FFmpeg y la codificación de vídeo después de que la salida de píxeles precodificada producida del precodificador profundo divulgado se llevara a cabo con un codificador HEVC configurado con control de velocidad de bits variable (VBR). Más allá de las realizaciones presentadas, los procedimientos descritos en el presente documento se pueden realizar con el rango completo de opciones y la adaptabilidad descritos en los ejemplos anteriores, y todas estas opciones y sus adaptaciones están cubiertas por esta divulgación.

La figura 10 muestra un procedimiento 1000 para preprocesar datos de imágenes usando una red de preprocesamiento que comprende un conjunto de pesos interconectados. El procedimiento 1000 se puede realizar mediante un dispositivo informático, de acuerdo con realizaciones. El procedimiento 1000 puede realizarse, al menos en parte, mediante hardware y/o software. El preprocesamiento se realiza antes de codificar los datos de imágenes preprocesados con un codificador externo. En el elemento 1010, los datos de imágenes de una o más imágenes se reciben en la red de preprocesamiento. Los datos de imágenes pueden recuperarse del almacenamiento (por ejemplo, en una memoria), o pueden recibirse de otra entidad. En el elemento 1020, los datos de imágenes se procesan usando la red de preprocesamiento (por ejemplo, aplicando los pesos de la red de preprocesamiento a los datos de imágenes) para generar una representación de píxeles de salida que codificar con el codificador externo. Los pesos de la red de preprocesamiento están entrenados para optimizar una combinación de: al menos una puntuación de calidad indicativa de la calidad de la representación de píxeles de salida; y una puntuación de velocidad indicativa de los bits requeridos por el codificador externo para codificar la representación de píxeles de salida. En realizaciones, el procedimiento 1000 comprende codificar la representación de píxeles de salida, por ejemplo usando el codificador externo. La representación de píxeles de salida codificada puede transmitirse, por ejemplo, a un dispositivo de visualización para su descodificación y posterior visualización.

Las realizaciones de la divulgación incluyen los procedimientos descritos anteriormente realizados en un dispositivo informático, tal como el dispositivo informático 1100 mostrado en la figura 11. El dispositivo informático 1100 comprende una interfaz de datos 1101, a través de la cual se pueden enviar o recibir datos, por ejemplo a través de una red. El dispositivo informático 1100 comprende además un procesador 1102 en comunicación con la interfaz de datos 1101, y una memoria 1103 en comunicación con el procesador 1102. De esta manera, el dispositivo informático 1100 puede recibir datos, tales como datos de imágenes o datos de vídeo, a través de la interfaz de datos 1101, y el procesador 1102 puede almacenar los datos recibidos en la memoria 1103 y procesarlos para realizar los procedimientos descritos en el presente documento, que incluyen preprocesar datos de imágenes antes de su codificación usando un codificador externo y, opcionalmente, codificar los datos de imágenes preprocesados.

Cada dispositivo, módulo, componente, máquina o función descrito en relación con cualquiera de los ejemplos descritos en el presente documento puede comprender un procesador y/o sistema de procesamiento o puede estar comprendido en un aparato que comprende un procesador y/o un sistema de procesamiento. Uno o más aspectos de las realizaciones descritas en el presente documento comprenden procesos realizados por un aparato. En algunos ejemplos, el aparato comprende uno o más sistemas de procesamiento o procesadores configurados para llevar a cabo estos procesos. En este sentido, las realizaciones pueden implementarse, al menos en parte, mediante software informático almacenado en memoria (no transitoria) y ejecutable por el procesador, o mediante hardware, o mediante una combinación de software y hardware almacenados de forma tangible (y firmware almacenado de forma tangible). Las realizaciones también se extienden a programas informáticos, particularmente programas informáticos en un medio portador, adaptados para poner en práctica las realizaciones descritas anteriormente. El programa puede estar en forma de código fuente no transitorio, código de objetos o en cualquier otra forma no transitoria adecuada para su uso en la implementación de procesos de acuerdo con realizaciones. El medio portador puede ser cualquier entidad o dispositivo capaz de portar el programa, tal como una RAM, una ROM, un dispositivo de memoria óptica, etc.

Se proporcionan diversas medidas (incluidos procedimientos, aparatos, dispositivos informáticos y productos de programa informático) para procesar datos de píxeles a partir de una única o una pluralidad de cuadros de imágenes

o vídeo usando un conjunto de pesos interconectados en una red que está configurada para convertir entradas en una representación de píxeles que minimice la combinación de los dos aspectos siguientes: (i) métricas objetivas que evalúan la distorsión de señal y puntuaciones que evalúan la pérdida de calidad estética o perceptiva, ya sea de manera independiente o en función de la imagen única o la pluralidad de imágenes de entrada;
(ii) una puntuación que representa la velocidad de bits por píxel (bpp) o bits por segundo (bps) necesaria para codificar la nueva representación de píxeles con un codificador externo de imágenes o vídeo que está diseñado para minimizar los bpp y mantener la fidelidad de imagen lo más alta posible de acuerdo con su propia puntuación de fidelidad de imagen.

En realizaciones, la resolución de los datos de píxeles aumenta o disminuye de acuerdo con una relación dada de escalado ascendente o descendente que puede ser un número entero o fraccionario y también incluye una relación de 1 (unidad) que no corresponde a ningún cambio de resolución. En realizaciones, la salida se mapea con una combinación lineal o no lineal de pesos, que están interconectados en una red y pueden incluir no linealidades tales como funciones de activación y capas de agrupación.

En realizaciones, la salida se corrompe para introducir una pérdida de fidelidad similar a la esperada por un codificador de imágenes o vídeo con pérdidas. Se puede hacer que el procedimiento de corrupción sean funciones matemáticamente diferenciables por aproximación de los operadores no diferenciables con una mezcla de los diferenciables y una aproximación adecuada.

En realizaciones, la representación de píxeles (opcionalmente escalados de manera ascendente o descendente) se redimensiona a la resolución de imagen o vídeo original usando un filtro lineal o no lineal y mediciones durante una fase de configuración o entrenamiento.

En realizaciones, las mediciones de la fase de configuración o de entrenamiento se usan para optimizar: (i) una puntuación de calidad que representa la opinión objetiva, perceptiva, estética o humana sobre la representación de píxeles redimensionados en la dirección de la reconstrucción o calidad visual mejorada; (ii) una puntuación de velocidad que representa los bits por píxel (bpp) o los bits por segundo (bps) necesarios para codificar la representación de píxeles con un codificador externo de imágenes o vídeo, en la dirección de una velocidad más baja.

En realizaciones, la combinación de puntuaciones de calidad y de velocidad bpp o bps se optimiza de acuerdo con un procedimiento de optimización lineal o no lineal que ajusta los pesos de las redes y el tipo de arquitectura usada para interconectarlos.

En realizaciones, el procedimiento de optimización lineal o no lineal es cualquier combinación de aprendizaje por retropropagación y actualizaciones de descenso de gradiente de pesos o errores calculados a partir de las puntuaciones utilizadas y las mediciones de fase de configuración o entrenamiento.

En realizaciones, representaciones individuales o grupos de nuevas representaciones de píxeles con una calidad y bpp o bps optimizados se pasan a un codificador posterior de imágenes o vídeo para codificarse y almacenarse en una memoria de ordenador o disco, o transmitirse a través de una red.

En realizaciones, el procedimiento de escalado descendente o ascendente es un filtro lineal o no lineal, o un procedimiento con capacidad de aprendizaje basado en datos y entrenamiento basado en retropropagación con procedimientos de descenso de gradiente.

En realizaciones, el codificador utilizado es un codificador de imágenes o vídeo basado en normas, tal como un codificador estándar ISO JPEG o ISO MPEG, o un codificador propietario o exento de regalías, tal como, pero no limitado a, un codificador AOMedia.

En realizaciones, se usa un filtro lineal, en donde el filtro puede ser un filtro de desenfoque o de mejora de bordes.

En realizaciones, se proporcionan pares de imágenes o vídeos de alta resolución y baja resolución y la imagen de baja resolución se escala de manera ascendente y optimiza para mejorar y/o hacer coincidir la calidad o velocidad con la imagen de alta resolución.

En realizaciones, la puntuación de calidad a minimizar incluye una o más de las siguientes puntuaciones de calidad de imagen objetiva, perceptiva o estética: relación máxima de señal/ruido, métrica de índice de similitud estructural (SSIM), métricas de calidad multiescala tales como la métrica de pérdida de detalle o SSIM multiescala, métricas basadas en múltiples puntuaciones de calidad y aprendizaje y entrenamiento basados en datos, tales como la fusión de evaluación multimétodo de vídeo (VMAF) o métricas de calidad estética, tales como las descritas por Deng, Y., Loy, C.C. y Tang, X., en su artículo: "Image aesthetic assessment: An experimental survey". *IEEE Signal Processing*

Magazine, 34(4), páginas 80-106, 2017, y variaciones de esas métricas.

En realizaciones, la puntuación que representa la velocidad de bpp o bps para codificar la nueva representación de píxeles se modela con un conjunto de ecuaciones que expresan la velocidad de bpp o bps esperada necesaria por un codificador estándar de imágenes o vídeo.

En realizaciones, la puntuación que representa la velocidad de bpp o bps para codificar la nueva representación de píxeles se entrena con procedimientos de retropropagación y descenso de gradiente y datos de entrenamiento que representan la velocidad de bpp o bps del codificador utilizado para comprimir la nueva representación de píxeles y la invención divulgada.

En realizaciones, la pluralidad de puntuaciones de calidad y la puntuación de velocidad de bpp o bps se combinan con pesos lineales o no lineales y estos pesos se entrenan en función de procedimientos de retropropagación y de descenso de gradiente con datos de entrenamiento representativos.

En realizaciones, el procedimiento de corrupción utilizado expresa la corrupción esperada de una transformada y cuantificación típicas basadas en bloques usadas en un codificador de imágenes o vídeo basado en bloques.

En realizaciones, el procedimiento de corrupción utilizado expresa la corrupción esperada de la transformada y la cuantificación de errores calculados a partir de un proceso típico de predicción temporal basado en bloques usado en un codificador de imágenes o vídeo basado en bloques.

En realizaciones, se hace que los procedimientos de corrupción usados sean funciones matemáticamente diferenciables, con parámetros que se entrenan con cualquier combinación de aprendizaje por retropropagación y actualizaciones de descenso de gradiente.

En realizaciones, el conjunto de ecuaciones que expresan la velocidad esperada de bps o bpp necesaria por un codificador de vídeo estándar para codificar una secuencia de vídeo puede incluir ambas velocidades para la codificación intercuadro e intracuadro, dependiendo del tipo de cuadro que se esté codificando.

En realizaciones, el entrenamiento de los procedimientos de calidad o velocidad, o el entrenamiento de los pesos de red para procesar los píxeles de entrada, o el entrenamiento de los procedimientos de corrupción se realizan a intervalos frecuentes o no frecuentes con nuevas mediciones de calidad, puntuaciones de velocidad de bpp e imágenes corruptas a partir de datos de imágenes codificados de codificadores externos de imágenes o vídeo, y los pesos, modelos o procedimientos de corrupción o funciones diferenciables actualizados reemplazan a los utilizados anteriormente.

Se proporcionan diversas medidas (que incluyen procedimientos, aparatos, dispositivos informáticos y productos de programa informático) para procesar datos de imágenes de una o más imágenes usando una red que comprende un conjunto de pesos interconectados, en donde la red está dispuesta para adquirir datos de imágenes de entrada y proporcionar una representación de píxeles, y está dispuesta además para minimizar: al menos una puntuación de calidad indicativa de la calidad de los datos de imágenes; y una puntuación de velocidad indicativa de los bits requeridos por un codificador de imágenes o vídeo para codificar la representación de píxeles de salida.

En realizaciones, la al menos una puntuación de calidad indica distorsión de señal en los datos de imágenes. En realizaciones, la al menos puntuación de calidad indica pérdida de calidad estética o perceptiva en los datos de imágenes.

En realizaciones, los bits requeridos por el codificador de imágenes o vídeo son bits por píxel o bits por segundo. En realizaciones, el codificador de imágenes o vídeo está dispuesto para minimizar bits por píxel. En realizaciones, el codificador de imágenes o vídeo está dispuesto para maximizar la fidelidad de imagen de acuerdo con una puntuación de fidelidad de imagen.

En realizaciones, las una o más imágenes son cuadros de vídeo.

En realizaciones, la resolución de la representación de píxeles aumenta o disminuye de acuerdo con una relación de escalado ascendente o descendente. En realizaciones, la relación de escalado ascendente o descendente es un número entero o un número fraccionario.

En realizaciones, la representación de píxeles está corrupta. En realizaciones, la etapa de corromper la representación de píxeles se realiza mediante una o más funciones matemáticamente diferenciables y una aproximación.

En realizaciones, la representación de píxeles se redimensiona a la resolución de los datos de imágenes de entrada. En realizaciones, el redimensionamiento se realiza mediante un filtro lineal o no lineal. En realizaciones, el filtro lineal o no lineal se configura durante una fase inicial de configuración o de entrenamiento.

En realizaciones, durante una fase inicial de configuración o entrenamiento, se optimiza lo siguiente: una puntuación de calidad que indica una opinión objetiva, perceptiva, estética o humana acerca de la representación de píxeles redimensionados, en la dirección de una reconstrucción o calidad visual mejoradas; y una puntuación de velocidad que indica los bits por píxel o bits por segundo necesarios para codificar la representación de píxeles mediante un codificador de imágenes o vídeo, en la dirección de una velocidad más baja.

En realizaciones, la combinación de la al menos una puntuación de calidad y una puntuación de velocidad se optimiza de acuerdo con un procedimiento de optimización lineal o no lineal que ajusta los pesos de la red. En realizaciones, la combinación de la al menos una puntuación de calidad y puntuación de velocidad se optimiza de acuerdo con un procedimiento de optimización lineal o no lineal que ajusta el tipo de la arquitectura usada para interconectar los pesos de la red. En realizaciones, el procedimiento de optimización lineal o no lineal es cualquier combinación de aprendizaje por retropropagación, actualizaciones de descenso de gradiente de pesos o errores calculados a partir de la al menos una puntuación de calidad y puntuación de velocidad, y mediciones de fase de configuración o entrenamiento.

En realizaciones, la representación de píxeles se codifica con un codificador de imágenes o vídeo. En realizaciones, el codificador de imágenes o vídeo es un codificador estándar ISO JPEG o ISO MPEG, o un codificador AOMedia.

En realizaciones, el escalado descendente o ascendente se realiza usando un filtro lineal o no lineal, o un procedimiento con capacidad de aprendizaje basado en datos y un entrenamiento basado en retropropagación con procedimientos de descenso de gradiente.

En realizaciones, la representación de píxeles se filtra usando un filtro lineal. En realizaciones, el filtro lineal es un filtro de desenfoque o de mejora de bordes.

En realizaciones, se proporcionan pares de imágenes o vídeos de alta resolución y baja resolución, y en donde la imagen de baja resolución se escala de manera ascendente y optimiza para mejorar y/o hacer coincidir la calidad o velocidad con la imagen de alta resolución.

En realizaciones, la al menos una puntuación de calidad incluye uno o más de lo siguiente: relación máxima de señal/ruido, métrica de índice de similitud estructural (SSIM), métricas de calidad multiescala, métrica de pérdida de detalle o SSIM multiescala, métricas basadas en múltiples puntuaciones de calidad y aprendizaje y entrenamiento basados en datos, fusión de evaluación multimétodo de vídeo (VMAF), métricas de calidad estética.

En realizaciones, la puntuación de velocidad se modela con un conjunto de ecuaciones que expresan la velocidad esperada necesaria por un codificador estándar de imágenes o vídeo. En realizaciones, la puntuación de velocidad se entrena con procedimientos de retropropagación y de descenso de gradiente y datos de entrenamiento que son representativos de la velocidad de un codificador utilizado para comprimir la representación de píxeles.

En realizaciones, la al menos una puntuación de calidad y la puntuación de velocidad se combinan con pesos lineales o no lineales, en donde los pesos lineales o no lineales se entrenan en función de procedimientos de retropropagación y de descenso de gradiente con datos de entrenamiento representativos.

En realizaciones, la representación de píxeles se corrompe para aproximarse a la corrupción esperada de una transformada y cuantificación típicas basadas en bloques usadas en un codificador de imágenes o vídeo basado en bloques. En realizaciones, la representación de píxeles se corrompe para aproximarse a la corrupción esperada de la transformada y cuantificación de errores calculados a partir de un proceso típico de predicción temporal basado en bloques usado en un codificador de imágenes o vídeo basado en bloques. En realizaciones, la corrupción se realiza usando funciones matemáticamente diferenciables con parámetros que se entrenan con una combinación de aprendizaje por retropropagación y actualizaciones de descenso de gradiente.

En realizaciones, los bits requeridos por un codificador de imágenes o vídeo para codificar la representación de píxeles de salida se determinan a partir de las velocidades de codificación intercuadro y/o intracuadro. En realizaciones, se usan velocidades de codificación intercuadro o intracuadro dependiendo del tipo de cuadro que se esté codificando.

En realizaciones, la al menos una puntuación de calidad, la puntuación de velocidad, los pesos de la red y/o los procedimientos de corrupción se entrenan, y en donde el entrenamiento se realiza a intervalos con nuevas mediciones de la al menos una puntuación de calidad, pesos de puntuación de velocidad y/o imágenes corruptas, respectivamente, según lo actualizado por el entrenamiento.

Si bien la presente divulgación se ha descrito e ilustrado con referencia a realizaciones particulares, los expertos en la técnica apreciarán que la divulgación se presta a muchas variaciones diferentes no ilustradas específicamente en el presente documento.

Cuando en la descripción anterior se mencionan números enteros o elementos que tienen equivalentes conocidos, obvios o previsibles, dichos equivalentes se incorporan en el presente documento como si se establecieran individualmente. Se debe hacer referencia a las reivindicaciones para determinar el verdadero alcance de la presente invención, que se debe interpretar para abarcar cualquiera de dichos equivalentes.

Bibliografía

[1] Dong, Jie, and Yan Ye. "Adaptive downsampling for high-definition video coding." IEEE Transactions on Circuits and Systems for Video Technology 24.3 (2014): 480-488.
[2] Douma, Peter, and Motoyuki Koike. "Method and apparatus for video upscaling." U.S. Patent No. 8,165,197. 24 Apr. 2012.
[3] Su, Guan-Ming, et al. "Guided image up-sampling in video coding." U.S. Patent No. 9,100,660. 4 Aug. 2015.
[4] Shen, Minmin, Ping Xue, and Ci Wang. "Downsampling based video coding using super-resolution technique." IEEE Transactions on Circuits and Systems for Video Technology21.6 (2011): 755-765.
[5] van der Schaar, Mihaela, and Mahesh Balakrishnan. "Spatial scalability for fine granular video encoding." U.S. Patent No. 6,836,512. 28 Dec. 2004.
[6] Boyce, Jill, et al. "Techniques for layered video encoding and decoding." U.S. Patent Application No. 13/738,138.
[7] Dar, Yehuda, and Alfred M. Bruckstein. "Improving low bit-rate video coding using spatio-temporal down-scaling." arXiv preprint arXiv:1404.4026 (2014).
[8] Martemyanov, Alexey, et al. "Real-time video coding/decoding." U.S. Patent No. 7,336,720. 26 Feb. 2008.
[9] Nguyen, Viet-Anh, Yap-Peng Tan, and Weisi Lin. "Adaptive downsampling/upsampling for better video compression at low bit rate." Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on. IEEE, 2008.
[10] Hinton, Geoffrey E., and Ruslan R. Salakhutdi- nov. "Reducing the dimensionality of data with neural networks." science313.5786 (2006): 504-507.
[11] van den Oord, Aaron, et al. "Conditional image generation with pixelcnn decoders." Advances in Neural Information Processing Systems. 2016.
[12] Theis, Lucas, et al. "Lossy image compression with compressive autoencoders." arXiv preprint arX- iv:1703. 00395(2017*).*
[13] Wu, Chao-Yuan, Nayan Singhal, and Philipp Krahenbuhl. "Video Compression through Image Interpolation." arXiv preprint arXiv: 1804.06919 (2018).
[14] Rippel, Oren, and Lubomir Bourdev. "Real-time adaptive image compression." arXiv preprint arx- iv:1705.05823 (2017).
[15] Golub, Gene H., and Charles F. Van Loan. Matrix computations. Vol. 3. JHU Press, 2012.
[16] Deng, Y., Loy, C.C. and Tang, X., "Image aesthetic assessment: An experimental survey," IEEE Signal Processing Magazine, 34(4), pp.80-106, 2017.

REIVINDICACIONES

1. Un procedimiento implementado por ordenador (1000) de preprocesamiento, antes de la codificación usando un codificador externo, de datos de imágenes usando una red de preprocesamiento, comprendiendo la red de preprocesamiento una red neuronal artificial que comprende un conjunto de pesos interconectados, comprendiendo el procedimiento:

recibir (1010), en la red de preprocesamiento, datos de imágenes de una o más imágenes; y procesar (1020) los datos de imágenes usando la red de preprocesamiento, aplicando los pesos de la red de preprocesamiento a los datos de imágenes para generar una representación de píxeles de salida, de la una o más imágenes, para su codificación con el codificador externo,
en donde los pesos de la red de preprocesamiento se entrenan usando retropropagación y descenso de gradiente para optimizar una combinación de:

al menos una puntuación de calidad indicativa de la calidad de la representación de píxeles de salida de las una o más imágenes; y
una puntuación de velocidad indicativa de los bits requeridos por el codificador externo para codificar la representación de píxeles de salida de las una o más imágenes, y
en donde la al menos una puntuación de calidad y la puntuación de velocidad se calculan usando un módulo de códec virtual que comprende una o más funciones matemáticamente diferenciables configuradas para emular un proceso de codificación, estando configurado el módulo de códec virtual para recibir, de la red de preprocesamiento, representaciones de píxeles de imágenes de entrada y para producir representaciones de píxeles distorsionadas de las imágenes de entrada usando las una o más funciones matemáticamente diferenciables, comprendiendo el módulo de códec virtual un componente de transformada de frecuencia, un componente de cuantificación y codificación y un componente de descuantificación y de transformada de frecuencia inversa.

2. Un procedimiento (1000) de acuerdo con la reivindicación 1, en donde la al menos una puntuación de calidad indica la distorsión de señal en la representación de píxeles de salida.

3. Un procedimiento (1000) de acuerdo con la reivindicación 1 o 2, en donde la al menos puntuación de calidad indica pérdida de calidad estética o perceptiva en la representación de píxeles de salida.

4. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, en donde la resolución de la representación de píxeles de salida aumenta o disminuye de acuerdo con una relación de escalado ascendente o descendente.

5. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, que comprende además la etapa de corromper la representación de píxeles de salida mediante la aplicación de una o más funciones matemáticamente diferenciables y una aproximación, en donde la representación de píxeles de salida está corrompida para aproximarse a la corrupción esperada de una transformada y una cuantificación basadas en bloques usadas en el codificador externo, y/o para aproximarse a la corrupción esperada de una transformada y cuantificación de errores calculados a partir de un proceso de predicción temporal basado en bloques usado en el codificador externo.

6. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, que comprende además la etapa de redimensionar la representación de píxeles de salida a la resolución de los datos de imágenes de entrada, usando un filtro lineal o no lineal configurado durante una fase inicial de configuración o entrenamiento.

7. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, que comprende además, durante una fase inicial de configuración o entrenamiento:

optimizar la al menos una puntuación de calidad en una dirección de reconstrucción o calidad visual mejoradas; y
optimizar la puntuación de velocidad en una dirección de velocidad más baja.

8. Un procedimiento (1000) de acuerdo con la reivindicación 7, en donde la al menos una puntuación de calidad y la puntuación de velocidad se optimizan de acuerdo con un procedimiento de optimización lineal o no lineal que ajusta los pesos de la red de preprocesamiento y/o ajusta el tipo de arquitectura usada para interconectar los pesos de la red de preprocesamiento.

9. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, que comprende además la etapa de codificar la representación de píxeles de salida con el codificador externo.

10. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, en donde el codificador externo es un codificador estándar ISO JPEG o ISO MPEG, o un codificador AOMedia.

11. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, que comprende filtrar la representación de píxeles de salida usando un filtro lineal, comprendiendo el filtro lineal un filtro de desenfoque o de mejora de bordes.

12. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, en donde la al menos una puntuación de calidad incluye uno o más de lo siguiente: relación máxima de señal/ruido, métrica de índice de similitud estructural (SSIM), métricas de calidad multiescala, métrica de pérdida de detalle o SSIM multiescala, métricas basadas en múltiples puntuaciones de calidad y aprendizaje y entrenamiento basados en datos, fusión de evaluación multimétodo de vídeo (VMAF), métricas de calidad estética.

13. Un procedimiento (1000) de acuerdo con cualquier reivindicación anterior, en donde la al menos una puntuación de calidad y la puntuación de velocidad se combinan con pesos lineales o no lineales, en donde los pesos lineales o no lineales se entrenan en función de procedimientos de retropropagación y de descenso de gradiente con datos de entrenamiento representativos.

14. Un dispositivo informático (1100), que comprende:

un procesador (1102); y
memoria (1103);
en donde el dispositivo informático (1100) está dispuesto para realizar, usando el procesador (1102), un procedimiento (1000) de acuerdo con cualquiera de las reivindicaciones 1 a 13.

15. Un producto de programa informático dispuesto, cuando se ejecuta en un dispositivo informático (1100) que comprende un procesador o memoria, para realizar un procedimiento (1000) de acuerdo con cualquiera de las reivindicaciones 1 a 13.

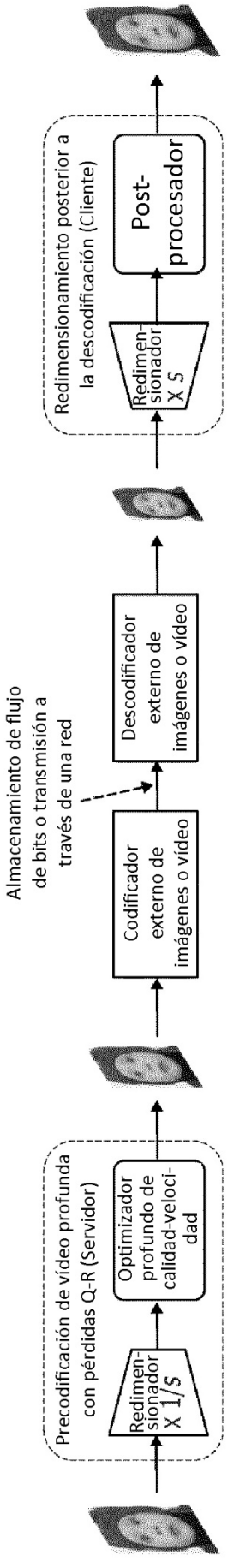

Precodificación de vídeo profunda con pérdidas Q-R (Servidor)
Redimen-sionador X 1/s
Optimizador profundo de calidad-veloci-dad
Codificador externo de imágenes o vídeo
Almacenamiento de flujo de bits o transmisión a través de una red
Descodificador externo de imágenes o vídeo
Redimensionamiento posterior a la descodificación (Cliente)
Redimen-sionador X s
Post-procesador

Figura 1

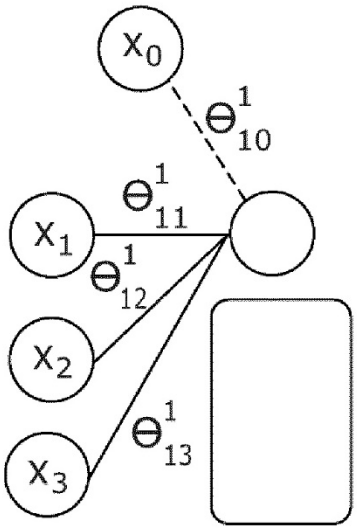


Figura 2 (a)

$$g(\Theta_{10}^{1}x_0 + \Theta_{11}^{1}x_1 + \Theta_{12}^{1}x_2 + \Theta_{13}^{1}x_3)$$

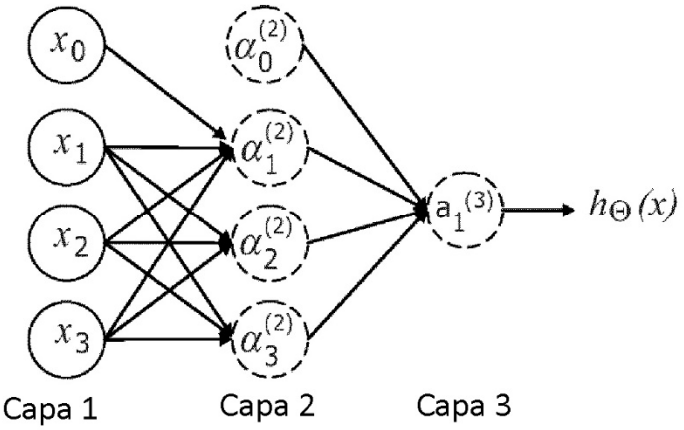
Θ_{13}^{1} significa:

1 (superíndice) – mapeo desde la capa 1

1 – mapeo al nodo 1 en capa 2 (L+1)

3 – mapeo desde el nodo 3 en capa 1 (L)

Figura 2 (b)



$$\alpha_1^{(2)} = g(\Theta_{10}^{(1)}x_0 + \Theta_{11}^{(1)}x_1 + \Theta_{12}^{(1)}x_2 + \Theta_{13}^{(1)}x_3)$$

$$\alpha_2^{(2)} = g(\Theta_{20}^{(1)}x_0 + \Theta_{21}^{(1)}x_1 + \Theta_{22}^{(1)}x_2 + \Theta_{23}^{(1)}x_3)$$

$$\alpha_3^{(2)} = g(\Theta_{30}^{(1)}x_0 + \Theta_{31}^{(1)}x_1 + \Theta_{32}^{(1)}x_2 + \Theta_{33}^{(1)}x_3)$$

$$h_\Theta(x) = \alpha_1^{(3)} = g(\Theta_{10}^{(2)}\alpha_0^{(2)} + \Theta_{11}^{(2)}\alpha_1^{(2)} + \Theta_{12}^{(2)}\alpha_2^{(2)} + \Theta_{13}^{(2)}\alpha_3^{(2)})$$

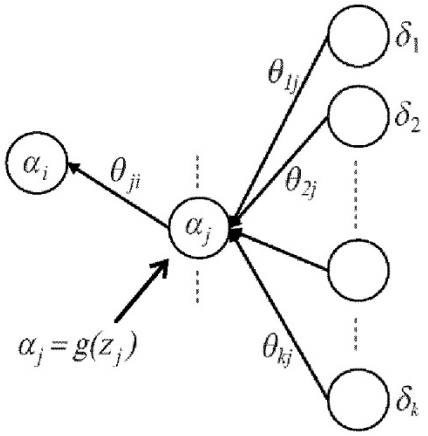


Figure 2(c)

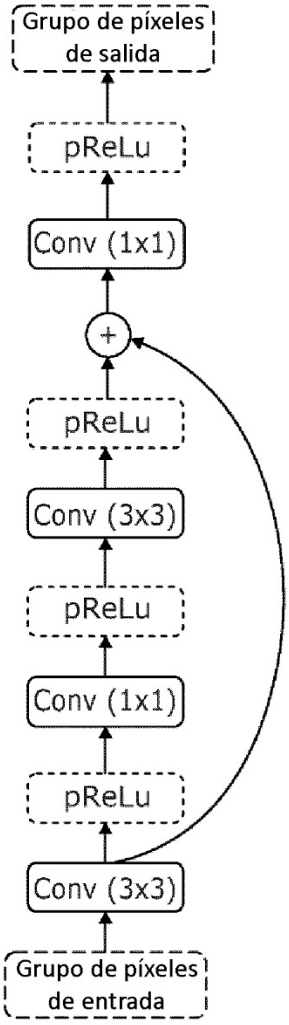


Figura 3

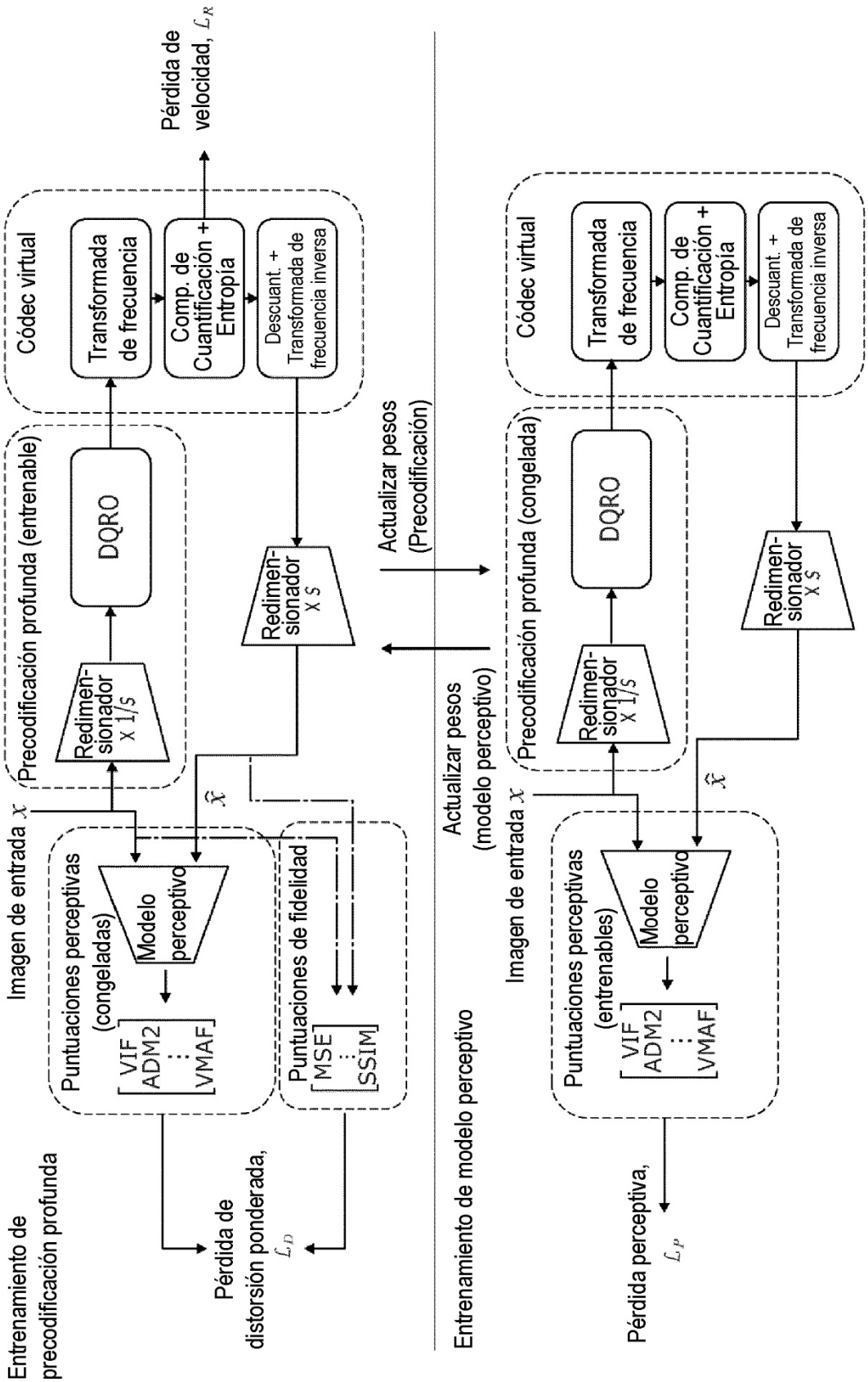

Entrenamiento de precodificación profunda
Imagen de entrada x
Precodificación profunda (entrenable)
Redimen-sionador x 1/s
DQRO
Códec virtual
Transformada de frecuencia
Comp. de Cuantificación + Entropía
Descuant. + Transformada de frecuencia inversa
Pérdida de velocidad, $\mathcal{L}_R$
Redimen-sionador x s
$\hat{x}$
Puntuaciones perceptivas (congeladas)
Modelo perceptivo
VIF ADM2 ⋮ VMAF
Puntuaciones de fidelidad
MSE ⋮ SSIM
Pérdida de distorsión ponderada, $\mathcal{L}_D$
Actualizar pesos (Precodificación)
Actualizar pesos (modelo perceptivo)
Entrenamiento de modelo perceptivo
Imagen de entrada x
Precodificación profunda (congelada)
Redimen-sionador x 1/s
DQRO
Códec virtual
Transformada de frecuencia
Comp. de Cuantificación + Entropía
Descuant. + Transformada de frecuencia inversa
Redimen-sionador x s
$\hat{x}$
Puntuaciones perceptivas (entrenables)
Modelo perceptivo
VIF ADM2 ⋮ VMAF
Pérdida perceptiva, $\mathcal{L}_P$

Figura 4

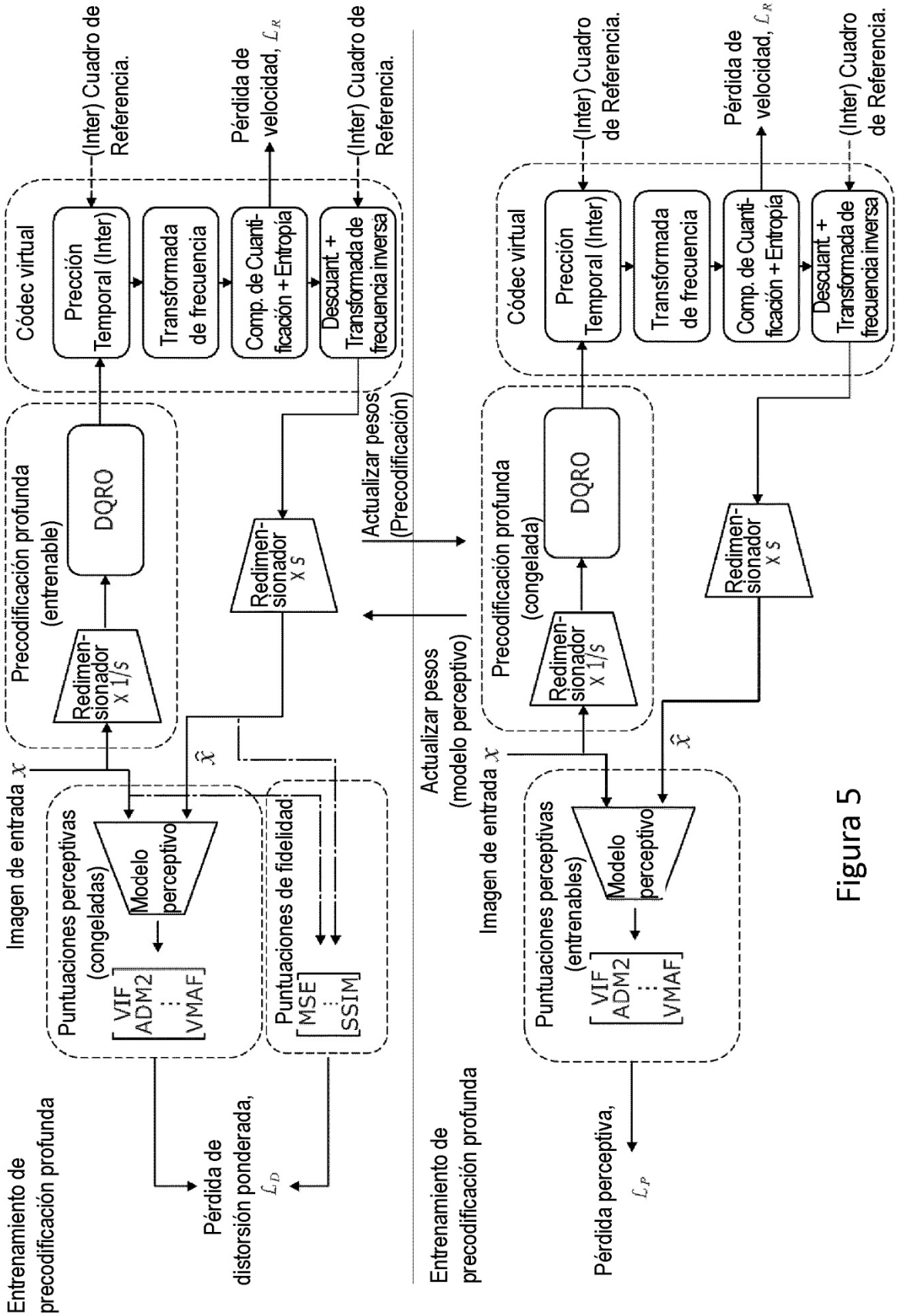

Entrenamiento de precodificación profunda
Imagen de entrada x
Precodificación profunda (entrenable)
Redimen-sionador x 1/S
DQRO
Códec virtual
Prección Temporal (Inter)
Transformada de frecuencia
Comp. de Cuanti-ficación + Entropía
Descuant + Transformada de frecuencia inversa
(Inter) Cuadro de Referencia.
Pérdida de velocidad, $\mathcal{L}_R$
Redimen-sionador x S
Actualizar pesos (Precodificación)
$\hat{x}$
Puntuaciones perceptivas (congeladas)
Modelo perceptivo
VIF ADM2 ... VMAF
Puntuaciones de fidelidad
MSE ... SSIM
Pérdida de distorsión ponderada, $\mathcal{L}_D$
Actualizar pesos (modelo perceptivo)
Entrenamiento de precodificación profunda
Imagen de entrada x
Precodificación profunda (congelada)
Redimen-sionador x 1/S
DQRO
Códec virtual
Prección Temporal (Inter)
Transformada de frecuencia
Comp. de Cuanti-ficación + Entropía
Descuant + Transformada de frecuencia inversa
(Inter) Cuadro de Referencia.
Pérdida de velocidad, $\mathcal{L}_R$
Redimen-sionador x S
$\hat{x}$
Puntuaciones perceptivas (entrenables)
Modelo perceptivo
VIF ADM2 ... VMAF
Pérdida perceptiva, $\mathcal{L}_P$

Figura 5

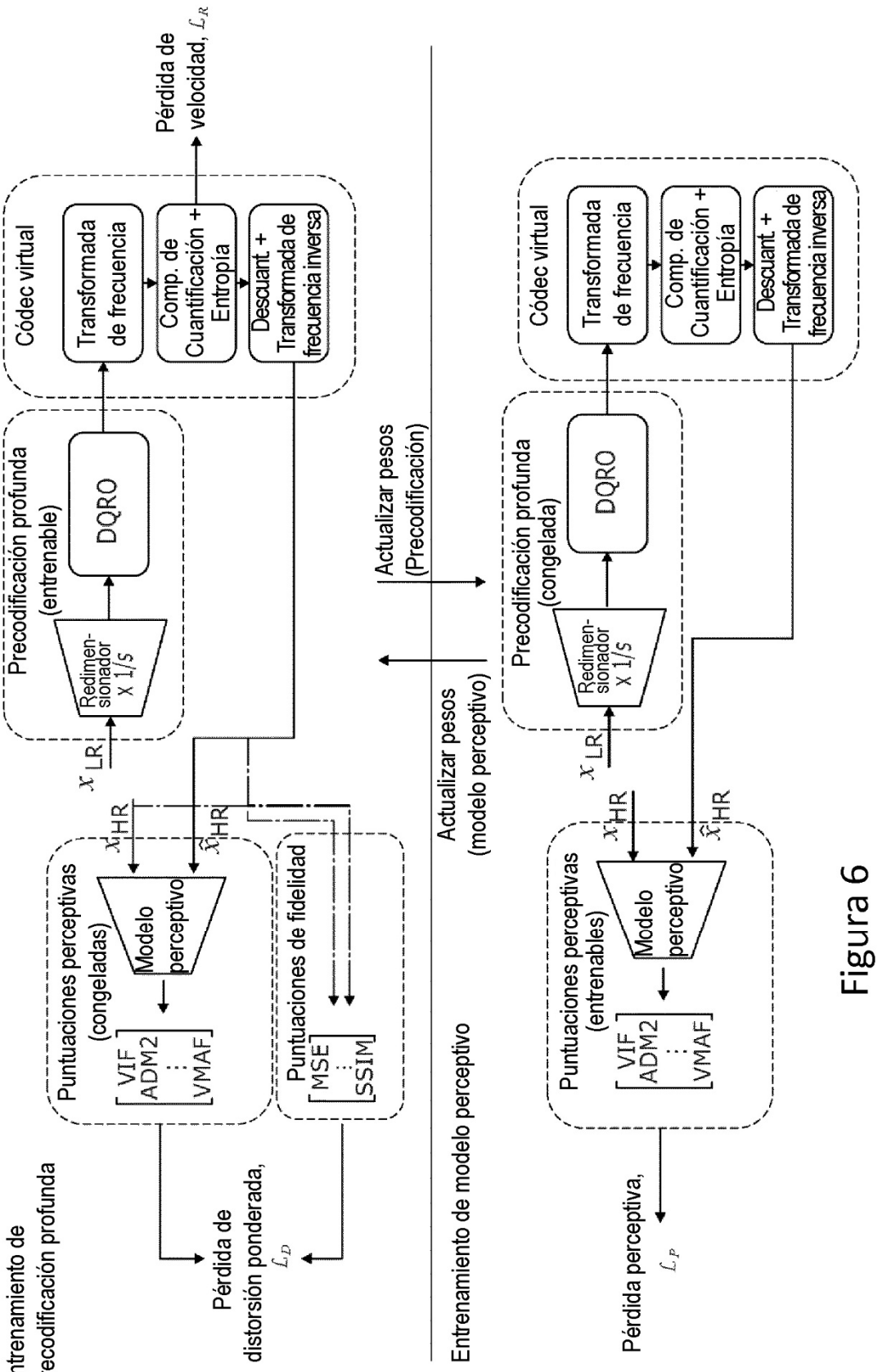

Entrenamiento de precodificación profunda
Pérdida de distorsión ponderada, L_D
Puntuaciones perceptivas (congeladas)
VIF ADM2 ... VMAF
Modelo perceptivo
Puntuaciones de fidelidad
MSE ... SSIM
x_{HR}
$\hat{x}_{HR}$
x_{LR}
Precodificación profunda (entrenable)
Redimen-sionador x 1/s
DQRO
Códec virtual
Transformada de frecuencia
Comp. de Cuantificación + Entropía
Descuant. + Transformada de frecuencia inversa
Pérdida de velocidad, L_R
Actualizar pesos (Precodificación)
Actualizar pesos (modelo perceptivo)
Entrenamiento de modelo perceptivo
Precodificación profunda (congelada)
Redimen-sionador x 1/s
DQRO
Códec virtual
Transformada de frecuencia
Comp. de Cuantificación + Entropía
Descuant. + Transformada de frecuencia inversa
x_{LR}
x_{HR}
$\hat{x}_{HR}$
Puntuaciones perceptivas (entrenables)
VIF ADM2 ... VMAF
Modelo perceptivo
Pérdida perceptiva, L_P

Figura 6

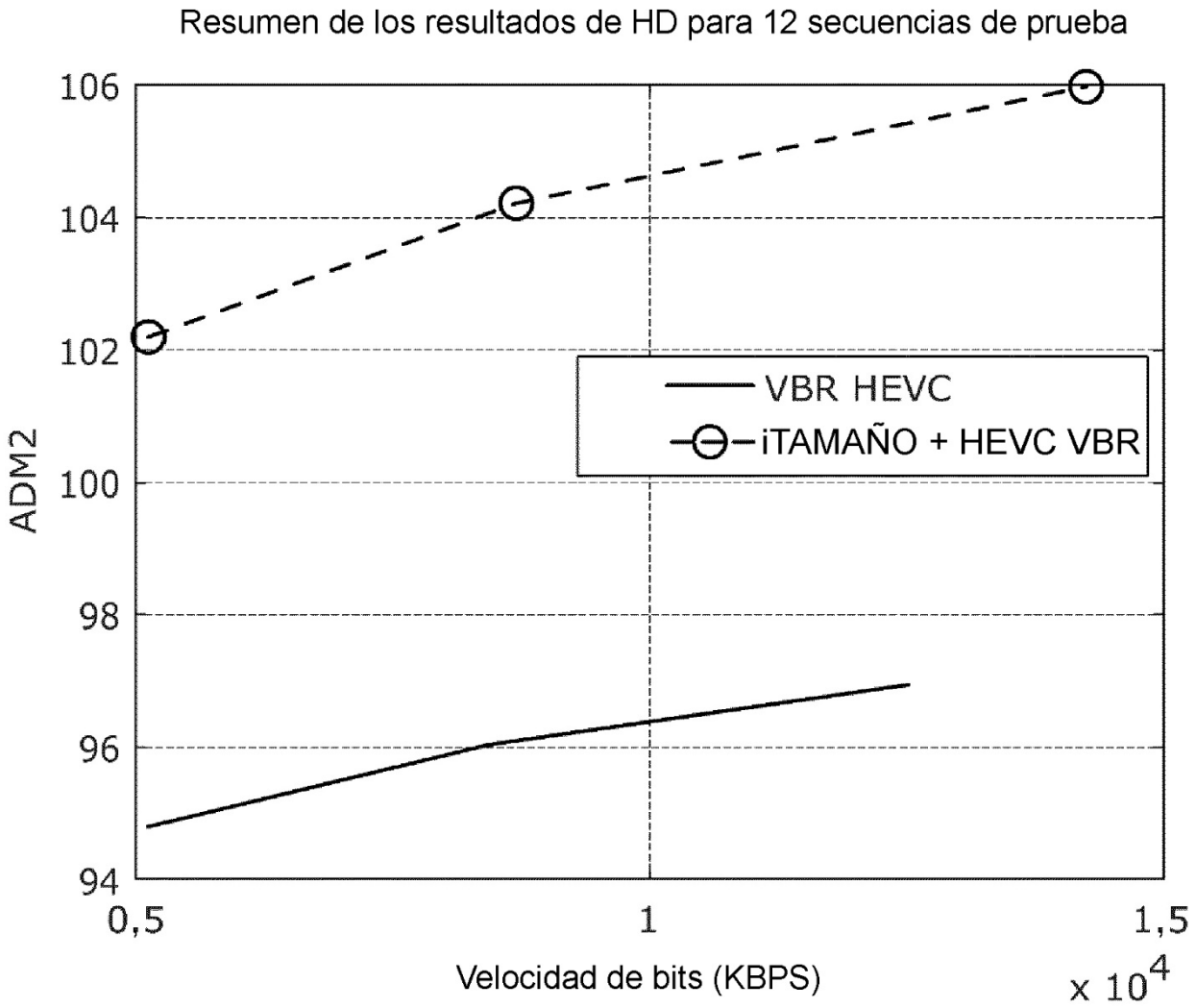

Resumen de los resultados de HD para 12 secuencias de prueba
106
104
102
100
98
96
94
ADM2
VBR HEVC
iTAMAÑO + HEVC VBR
0,5
1
1,5
Velocidad de bits (KBPS)
x 10^4

Figura 7

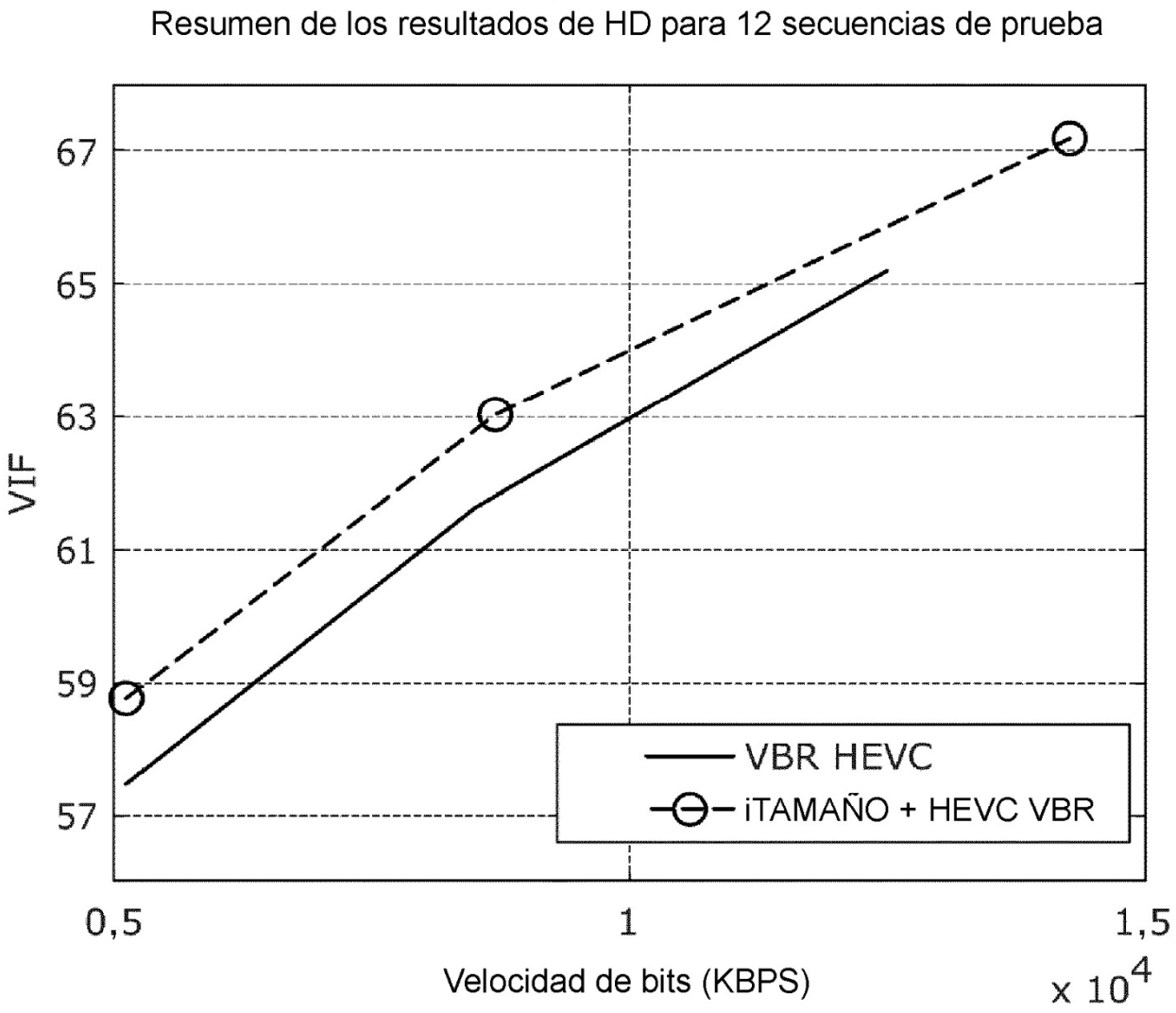

Resumen de los resultados de HD para 12 secuencias de prueba
VIF
67
65
63
61
59
57
0,5
1
1,5
Velocidad de bits (KBPS)
x 10^4
VBR HEVC
iTAMAÑO + HEVC VBR

Figura 8

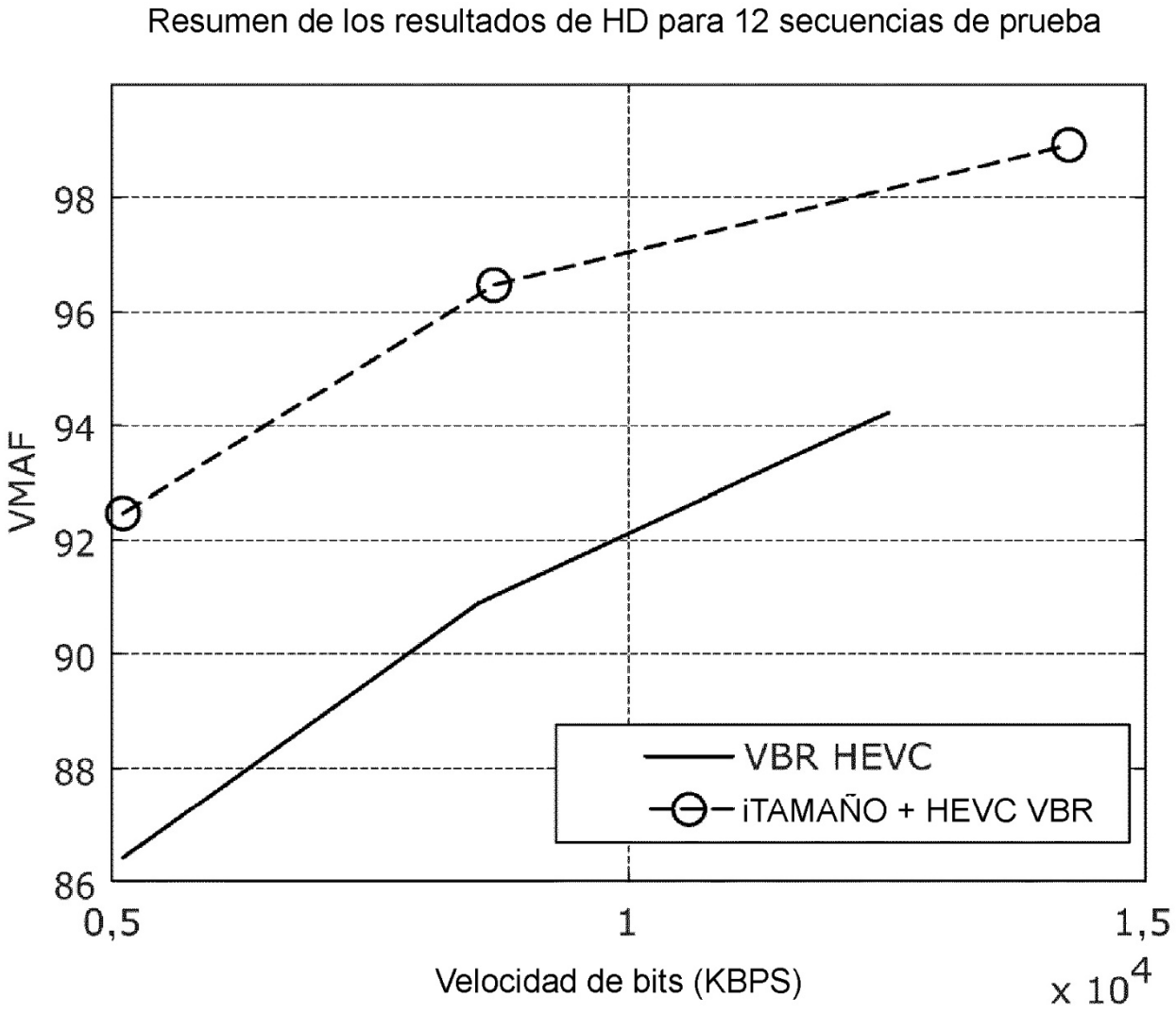

Resumen de los resultados de HD para 12 secuencias de prueba
VMAF
98
96
94
92
90
88
86
0,5
1
1,5
Velocidad de bits (KBPS)
x 10^4
VBR HEVC
iTAMAÑO + HEVC VBR

Figura 9

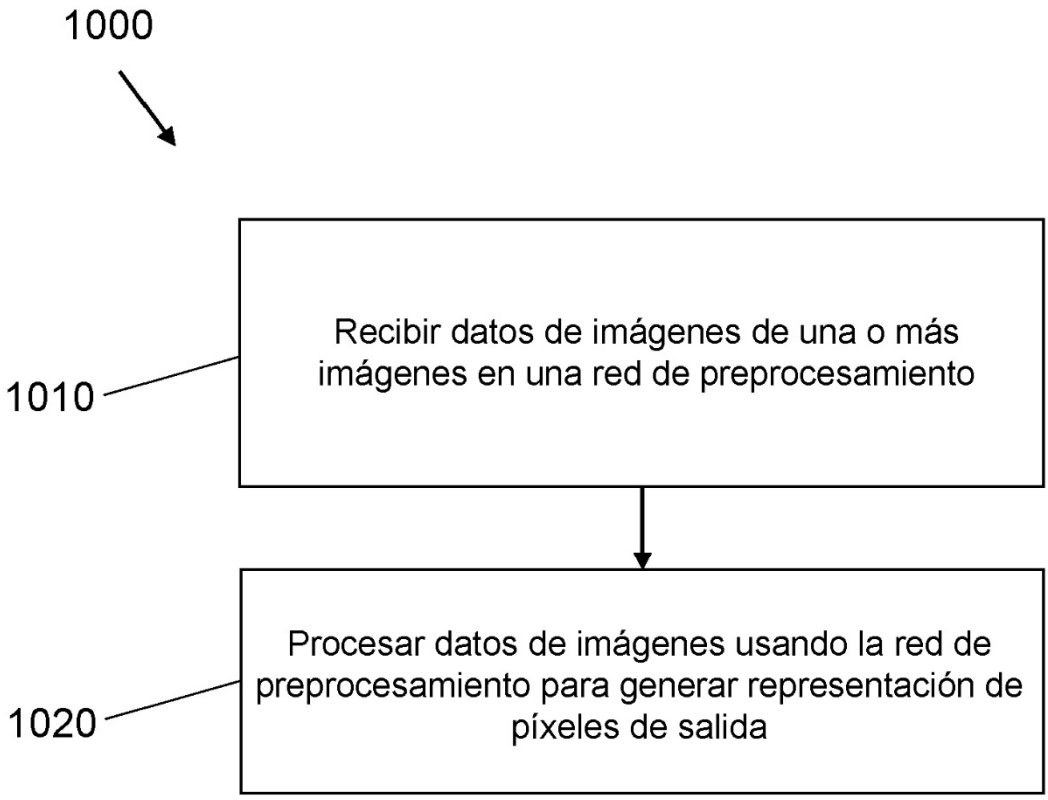

1000
Recibir datos de imágenes de una o más imágenes en una red de preprocesamiento
1010
Procesar datos de imágenes usando la red de preprocesamiento para generar representación de píxeles de salida
1020

Fig. 10

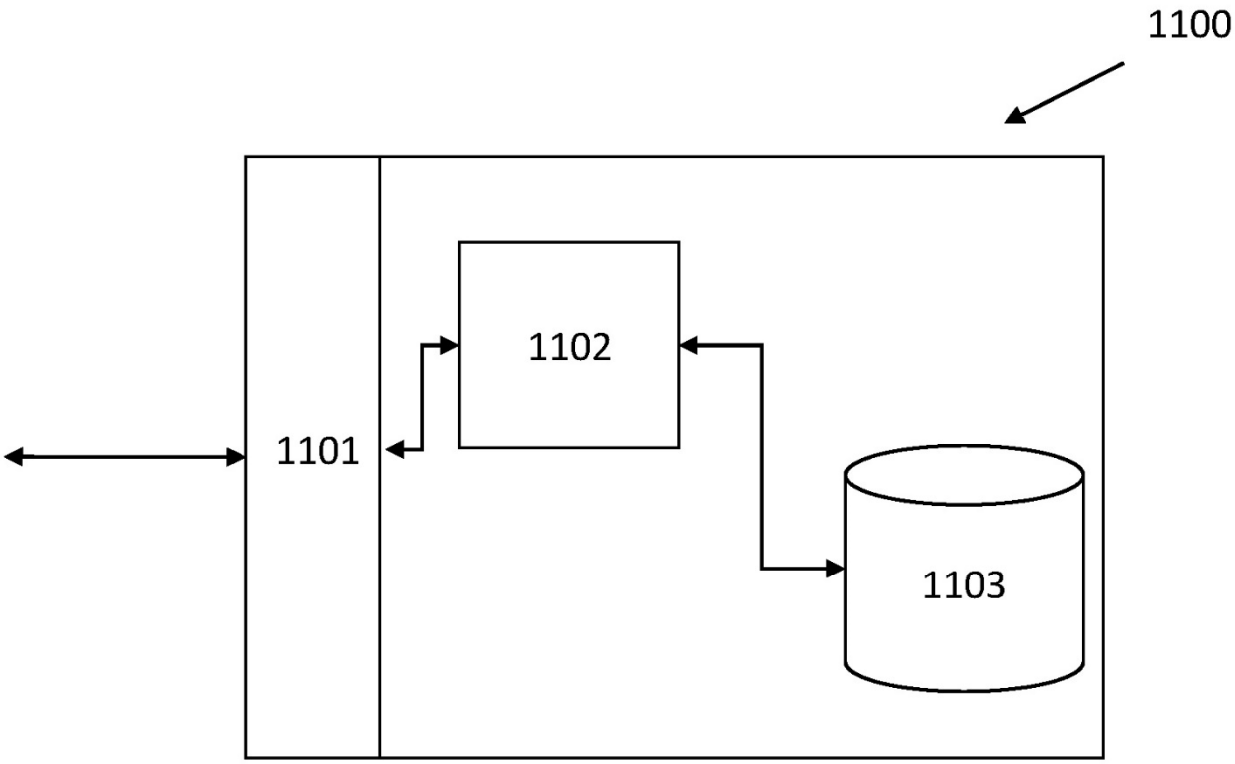


Fig. 11