(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 October 2016 (06.10.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/156864 A1

(51) International Patent Classification:
H04N 19/46 (2014.01) *H04N 19/86* (2014.01)
G06T 3/40 (2006.01) *H04N 19/59* (2014.01)
G06T 5/00 (2006.01) *H04N 19/147* (2014.01)
H04N 19/117 (2014.01) *H04N 19/172* (2014.01)
H04N 19/142 (2014.01) *H04N 19/136* (2014.01)
H04N 19/154 (2014.01) *H04N 19/19* (2014.01)
H04N 19/177 (2014.01) *H04N 19/85* (2014.01)
H04N 19/80 (2014.01) *G06T 9/00* (2006.01)

(21) International Application Number:
PCT/GB2016/050922

(22) International Filing Date:
31 March 2016 (31.03.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

1505544.5	31 March 2015 (31.03.2015)	GB
1507141.8	27 April 2015 (27.04.2015)	GB
1508742.2	21 May 2015 (21.05.2015)	GB
1511231.1	25 June 2015 (25.06.2015)	GB
1519425.1	3 November 2015 (03.11.2015)	GB
1519687.6	6 November 2015 (06.11.2015)	GB
PCT/GB2016/050423	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050430	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050424	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050426	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050432	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050431	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050425	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050427	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050428	19 February 2016 (19.02.2016)	GB
PCT/GB2016/050429	19 February 2016 (19.02.2016)	GB
1603144.5	23 February 2016 (23.02.2016)	GB
1604345.7	14 March 2016 (14.03.2016)	GB
1604672.4	18 March 2016 (18.03.2016)	GB

(71) Applicant *(for all designated States except US)*: **MAGIC PONY TECHNOLOGY LIMITED** [GB/GB]; Flat 2, 38 Gratton Road, London, Greater London W14 0JX (GB).

(72) Inventors; and
(71) Applicants *(for US only)*: **WANG, Zehan** [GB/GB]; 38 Gratton Road, Flat 2, London W14 0JX (GB). **BISHOP, Robert David** [GB/GB]; 38 Gratton Road, Flat 2, London W14 0JX (GB). **HUSZAR, Ferenc** [HU/GB]; 67 Cavendish Road, Cambridge CB1 3AE (GB). **THEIS, Lucas** [DE/GB]; 259 Kennington Lane, London SE11 5QU (GB).

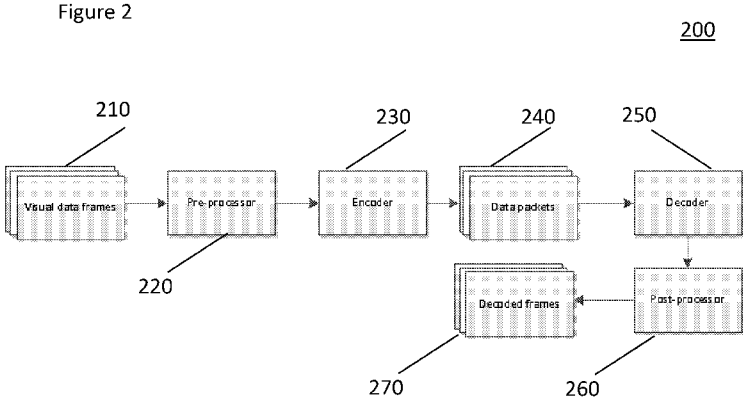
(74) Agent: **MATHYS & SQUIRE LLP**; The Shard, 32 London Bridge Street, London SE1 9SG (GB).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC,

[Continued on next page]

(54) Title: TRAINING END-TO-END VIDEO PROCESSES

Figure 2



(57) Abstract: The present invention relates to a method for training a plurality of visual processing algorithms for processing visual data. Thee method comprising the steps of using a pre-processing hierarchical algorithm to process the visual data prior to encoding the visual data in visual data processing, and using a post-processing hierarchical algorithm to further process the visual data following decoding visual data in visual data processing. The steps of steps of encoding and decoding are performed with respect to a predetermined visual data codec and in some embodiments may be content specific.

WO 2016/156864 A1

SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

TRAINING END-TO-END VIDEO PROCESSES

Field of Invention

The present invention relates to methods and systems to process visual data. Specifically, the present invention relates to methods and systems to preserve visual information, and optimize efficiency during compression and decompression.

Background

Quality of video and display technology

Developments in display technology have led to significant improvements in the resolution able to be displayed on display hardware, such as televisions, on computer monitors, and using video projectors. For example, television screens that are able to display 'High Definition' or 'HD' resolution content (typically having a resolution of 1920 x 1080 pixels) have been broadly adopted by consumers. More recently, television screens able to display Ultra High Definition or 'Ultra HD' resolution content (typically having a resolution over 3840 x 2160 pixels) are starting to become more widespread.

In contrast, HD resolution video content is only now becoming commonplace and most legacy content is only available as either Digital Versatile Disc Video (or 'DVD-Video') resolution (typically having a resolution of 720 x 586 pixels of 720 x 480 pixels) or Standard Definition or 'SD' Resolution (where the video content only has a resolution of 640 x 480 pixels). Some broadcast channels are limited to SD resolutions. Video-streaming services can be restricted to operating at DVD-Video or SD resolutions, to reduce transmission problems where consumers have limitations on available transmission bandwidths or because of a lack of legacy content at higher resolutions.

As a result, there can be a lack of sufficiently high-resolution video content for display on HD and Ultra HD television screens, for both current video content as well as for legacy video content and video streaming services. Also, over time mobile

devices such as mobile 'phones and tablet computers with increasingly larger and higher-resolution screens are being produced and adopted by users. Further, current video content, being output at HD resolutions, is already at a significantly lower resolution than can be displayed by the latest consumer displays operating at, for example, Ultra HD resolutions. To provide sufficiently immersive virtual reality (or "VR") experiences, display technology needs to be sufficiently high resolution even for smaller screen sizes.

The user experience of having to display content that has significantly lower resolution than the user's default screen/display resolution is not optimal.

Growth in data transmission and network limitations

The amount of visual data being communicated over data networks such as the Internet has grown dramatically over time and there is increasing consumer demand for high-resolution, high quality, high fidelity visual data content, such as video streaming including, for example, video at HD and Ultra HD resolution. As a result, there are substantial challenges in meeting this growing consumer demand and high performance video compression is required to enable efficient use of existing network infrastructure and capacity.

Video data already makes up a significant fraction of all data traffic communicated over the Internet, and mobile video (i.e. video transmitted to and from mobile devices over wireless data networks such as UTMS/CDMA) is predicted to increase 13-fold between 2014 and 2019, accounting for 72 percent of total mobile data traffic by the end of that forecast period. As a result, there are substantial challenges in meeting this growing consumer demand and more efficient visual data transmission is required to enable efficient use of existing network infrastructure and capacity.

To stream video to consumers using available streaming data bandwidth, media content providers can down-sample or transcode the video content for transmission over a network at one or a variety of bitrates so that the resolution of the video can be appropriate for the bitrate available over each connection or to each

device and correspondingly the amount of data transferred over the network can be better matched to the available reliable data rates. For example, a significant proportion of current consumer Internet connections are not able to reliably support continuous streaming of video at an Ultra HD resolution, so video needs to be streamed at a lower quality or lower resolution to avoid buffering delays.

Further, where a consumer wishes to broadcast or transmit video content, the uplink speeds of consumer Internet connections are typically a fraction of the download speeds and thus only lower quality or lower resolution video can typically be transmitted. In addition, the data transfer speeds of typical consumer wireless networks are another potential bottleneck when streaming video data for video at resolutions higher than HD resolutions or virtual reality data and content to/from contemporary virtual reality devices. A problem with reducing the resolution of a video when transmitting it over a network is that the reduced resolution video may not be at the desired playback resolution, but in some cases there is either not sufficient bandwidth or the bandwidth available is not reliable during peak times for transmission of a video at a high resolution.

Alternatively, even without reducing the original video resolution, the original video may have a lower resolution than desired for playback and so may appear at a suboptimal quality when displayed on higher-resolution screens as a result of quantisation and / or compression as well as lowering resolutions. This is particular apparent on mobile devices, such as cell phones and or tablet computers, wherein the bandwidth availability and the cost of data for networks impact the end-user. Similarly, bandwidth and cost implications may affect fixed-line connections in developing countries.

Video Compression Techniques

Existing commonly used video compression techniques, such as H.264 and VP8, as well as proposed techniques, such as H.265, HEVC and VP9, all generally use similar approaches and families of compression techniques. These compression techniques make a trade-off between the quality and the bit-rate of video data streams when providing inter-frame and intra-frame compression, but the amount of

compression possible is largely dependent on the image resolution of each frame and the complexity of the image sequences.

To illustrate the relationship between bitrate and resolution among other factors, it is possible to use an empirically-derived formula to show how the bitrate of a video encoded with, for example the H.264 compression technique, relates to the resolution of that video:

$$\text{bitrate} \propto Q \times w \times h \times f \times m$$

where Q is the quality constant, w is the width of a video, h is the height of a video, f is the frame-rate of a video and m is the motion rank, where $m \in \{1,\ldots,4\}$ and a higher m is used for fast-changing hard-to-predict content.

The above formula illustrates the direct relationship between the bitrate and the quality constant Q. A typical value for a H.264 codec, for example, that could be selected for Q would be 0.07 based on published empirical data, but a significant amount of research is directed to optimising a value for Q. The skilled person will appreciate other values of Q may be appropriate, especially when using different codecs.

The above formula also illustrates the direct relationship between the bitrate and the complexity of the image sequences, i.e. variable m. The aforementioned existing video codecs focus on spatial and temporal compression techniques. The newer proposed video compression techniques, such as H.265, HEVC and VP9, seek to improve upon the motion prediction and intra-frame compression of previous techniques, i.e. optimising a value for m.

The above formula further illustrates a direct relationship between the bitrate, the resolution of the video, and the quantisation value, Q, i.e. variables w and h. In order to reduce the resolution of video, modify the quantisation and the arrangement of data, several techniques exist to optimise video data to reduce the bitrate.

As a result of the disadvantages of current compression approaches, existing network infrastructure and video streaming mechanisms are becoming increasingly

inadequate to deliver large volumes of high quality video content to meet ever-growing consumer demands for this type of content. This can be of particular relevance in certain circumstances, for example in relation to live broadcasts, where bandwidth is often limited, and extensive processing and video compression cannot take place at the location of the live broadcast without a significant delay due to inadequate computing resources being available at the location.

Machine Learning Techniques

Machine learning is the field of study where a computer or computers learn to perform classes of tasks using the feedback generated from the experience or data gathered that the machine learning process acquires during computer performance of those tasks.

Typically, machine learning can be broadly classed as supervised and unsupervised approaches, although there are particular approaches such as reinforcement learning and semi-supervised learning which have special rules, techniques and/or approaches.

Supervised machine learning is concerned with a computer learning one or more rules or functions to map between example inputs and desired outputs as predetermined by an operator or programmer, usually where a data set containing the inputs is labelled.

Unsupervised learning is concerned with determining a structure for input data, for example when performing pattern recognition, and typically uses unlabelled data sets.

Reinforcement learning is concerned with enabling a computer or computers to interact with a dynamic environment, for example when playing a game or driving a vehicle.

Various hybrids of these categories are possible, such as “semi-supervised” machine learning where a training data set has only been partially labelled.

For unsupervised machine learning, there is a range of possible applications such as, for example, the application of computer vision techniques to image processing or video enhancement. Unsupervised machine learning is typically applied to solve problems where an unknown data structure might be present in the data. As the data is unlabelled, the machine learning process is required to operate to identify implicit relationships between the data for example by deriving a clustering metric based on internally derived information. For example, an unsupervised learning technique can be used to reduce the dimensionality of a data set and attempt to identify and model relationships between clusters in the data set, and can for example generate measures of cluster membership or identify hubs or nodes in or between clusters (for example using a technique referred to as weighted correlation network analysis, which can be applied to high-dimensional data sets, or using k-means clustering to cluster data by a measure of the Euclidean distance between each datum).

Semi-supervised learning is typically applied to solve problems where there is a partially labelled data set, for example where only a subset of the data is labelled. Semi-supervised machine learning makes use of externally provided labels and objective functions as well as any implicit data relationships.

When initially configuring a machine learning system, particularly when using a supervised machine learning approach, the machine learning algorithm can be provided with some training data or a set of training examples, in which each example is typically a pair of an input signal/vector and a desired output value, label (or classification) or signal. The machine learning algorithm analyses the training data and produces a generalised function that can be used with unseen data sets to produce desired output values or signals for the unseen input vectors/signals. The user needs to decide what type of data is to be used as the training data, and to ensure there is enough training data to optimise the learning of the function, or to ensure the function is able to predict an optimal output. Furthermore, the user must however take care to ensure that the training data contains enough information to accurately predict desired output values without providing too many features (which can result in too many dimensions being considered by the machine learning

process during training, and could also mean that the machine learning process does not converge to good solutions for all or specific examples). In some embodiments, instead of restricting the amount of information in the training data, regularisation or Bayesian methods may be used. The user must also determine the desired structure of the learned or generalised function, for example whether to use support vector machines or decision trees.

The use of unsupervised or semi-supervised machine learning approaches are sometimes used when labelled data is not readily available, or where the system generates new labelled data from unknown data given some initial seed labels.

Current training approaches for most machine learning algorithms can take significant periods of time, which delays the utility of machine learning approaches and also prevents the use of machine learning techniques in a wider field of potential application.

Machine learning (or other learned approaches) may be incorporated into some super resolution techniques, to improve the effectiveness of such techniques.

For example, one machine learning approach that can be used for image enhancement, using dictionary representations for images, is a technique generally referred to as dictionary learning. This approach has shown effectiveness in low-level vision tasks like image restoration.

Summary of Invention

Aspects and/or embodiments are set out in the appended claims. Some aspects and/or embodiments can optimise the effectiveness of visual or content representations using machine learning techniques. These and other aspects and embodiments are also described herein.

Certain aspects and/or embodiments seek to provide techniques for generating algorithms that can be used to enhance visual data based on received input visual data and a plurality of pieces of training data.

Other aspects and/or embodiments seek to provide techniques for machine learning.

According to a first aspect of the invention there is provided a method for training a plurality of visual processing algorithms for processing visual data, the method comprising the steps of: using a pre-processing hierarchical algorithm to process visual data prior to encoding the visual data in visual data processing; and using a post-processing hierarchical algorithm to reconstruct visual data following decoding visual data in visual data processing wherein the steps of encoding and decoding are performed with respect to a predetermined visual data codec.

The training of the algorithms enables a more efficient encoding of the visual data to be transmitted, maintain the quality of the visual data and also ensuring a suitable size for transmission over a network by producing a bit-stream which is of lower quality than the input visual data.

Optionally, in some embodiments, the method may further comprise the step of receiving one or more sections of visual data.

Receiving the visual data enables sections of the visual data to be stored remoted, for example on the Internet. Alternatively, the visual data may be stored locally on a device configured to perform the method.

Optionally, processing the visual data comprises optimising the visual data.

In some embodiments, the processing of the visual data may also be used to optimise the visual data such as reduce the size and/or enhance the quality. For example, in some embodiments this may include increasing the resolution whilst maintaining a suitable size for transmission.

Optionally, one or more parameters associated with the pre-processing hierarchical algorithm may be stored in a library for re-use in encoding alternative visual data similar to the visual data used for training, or may be transmitted to a device configured to process alternative visual data similar to the visual data used for training. Similarly, one or more parameters associated with the post-processing

hierarchical algorithm, may be transmitted with any process visual data to a remote device.

The storing of parameters associated with the trained hierarchical algorithms used to pre- and post- process the visual data enables similar training techniques to be used on similar visual data. Furthermore, it allows any parameters to be transmitted with the visual data to a remote device for displaying, removing the need to re-train the hierarchical algorithms on said remote device.

Optionally, the pre-processing and/or post-processing hierarchical algorithm may comprise a layer that generalises the visual data processing, and may further comprise a layer that generalises the encoding and/or decoding performed during visual data processing.

The generalisation of the visual data processing and the encoding or decoding means the algorithm may analyse training data and produce a generalised function that may be used with unseen data sets to produce desired output values or signals for the unseen input vectors/signals. The user needs to decide what type of data is to be used as the training data, and ensure there is enough training data to optimise the learning of the function, or to ensure the function is able to predict an optimal output. Furthermore, the user must however take care to ensure that the training data contains enough information to accurately predict desired output values without providing too many features (which can result in too many dimensions being considered by the machine learning process during training, and could also mean that the machine learning process does not converge to good solutions for all or specific examples). In some embodiments, instead of restricting the amount of information in the training data, regularisation or Bayesian methods may be used. The user must also determine the desired structure of the learned or generalised function, for example whether to use support vector machines or decision trees.

Optionally, the method may further comprise the step of receiving a plurality of criteria upon which to base any processing. In some embodiments, the criteria may be a specific bit rate or a quality characteristic.

The use of criteria enables the method to optimise and train any algorithms based on specific hardware, software, content type, or user requirements. In some embodiments, the specification of a bit-rate means the method is to optimise the algorithms such that the best quality visual data is produced for a specific bandwidth. Alternatively, some form of quality criteria, such as resolution, peak signal-to-noise ratio (PSNR), or mean squared error (MSE) or a perceptual or subjective metric, may be provided which would enable the method to optimise the algorithms so that the best compression is achieved whilst maintaining the specified quality.

Optionally, the hierarchical algorithms comprise a plurality of connected layers, and the connected layers may be sequential, recurrent, recursive, branching or merging.

Optionally, the visual data comprises one or more sections of visual data, and at least one section lower-quality visual data may comprise any of: a single frame of lower-quality visual data, a sequence of frames of lower-quality visual data, and a region within a frame or sequence of frames of lower-quality visual data. Furthermore, the lower-quality visual data may comprise a plurality of frames of video. In some embodiments, the visual data may comprise a plurality of frames of video, or a plurality of images.

Depending on the visual data being processed, in some embodiments models can be selected for sections of visual data comprising a sequence of frames or a region within a frame or sequence of frames. In these embodiments each could be necessary in order to provide the most efficient method of processing the original visual data.

Optionally, the hierarchical algorithm differs for each section of visual data.

In some embodiments, the use of different hierarchical algorithms for each section of visual data enables the most efficient hierarchical algorithm to be used for a particular section as opposed to using a single hierarchical algorithm for the entire visual data.

Optionally, the hierarchical algorithm is selected from a library of algorithms.

In some embodiments, a stored library of algorithms allows selection of a hierarchical algorithm for comparison without having to develop them or obtain them from an external source. In some embodiments, the comparison can be between a plurality of algorithms in the library. Use of such a library, in at least some embodiments, may result in the faster selection of a suitable hierarchical algorithm for enhancing the visual data or, in some embodiments, the most suitable hierarchical algorithm in a library (for example, by basing a measure of suitability on a predetermined metric).

Optionally, the standardised features of the at least one section of received lower-quality visual data are extracted and used to select the hierarchical algorithm from the library of algorithms.

In some embodiments, extracted standardised features are used to produce a value or series of values based on a metric from the input data. In these embodiments, the metric can then be used to select the pre-trained model from the library which is most appropriate for the input data, as each model in the library has associated metric values based on the input data from which the models were respectively trained, the selection based on the similarity between the metrics associated with the input data and each of the pre-trained models.

Optionally, the hierarchical algorithm to be selected from the library of algorithms is based on generating the highest quality version of the lower-quality visual data, preferably wherein quality can be defined by any of: an error rate; a bit error rate; a peak signal-to-noise ratio; or a structural similarity index.

The predetermined metrics used in some embodiments to determine the hierarchical algorithm to be selected can be based on a predicted quality of the output data for each pre-trained model. In some of these embodiments, quality can be defined by any or all of: an error rate; a peak signal-to-noise ratio; or a structural similarity index.

Optionally, the hierarchical algorithms are developed using a learned approach.

In some embodiments, hierarchical or non-hierarchical algorithms can be substantially accurate and therefore enable a more accurate reconstruction, for example produce higher quality visual data from the low-quality visual data that is transmitted, for example where quality can be measured by resolution, PSNR, MSE, or a perceptual measure or metric determining that the quality is sufficiently aesthetically pleasing or by a low reproduction error rate in comparison to the original high-quality visual data. In another example, the hierarchical or non-hierarchical algorithms can produce higher quality versions of visual data using the fidelity data. In some optional embodiments, a down-sampled version of the resulting visual data comes out to be the same or similar as a down-sampled version of the original visual data. In some embodiments, using a learned approach can substantially tailor the hierarchical model or models for each portion of visual data.

Optionally, the learned approach comprises machine learning techniques. The hierarchical algorithm may also be a non-linear hierarchical algorithm which may comprise one or more convolutional neural networks.

In some embodiments, through using a learning-based approach, i.e. an approach that does not rely on pre-defined visual data features and operators, the model(s) can be optimised for each section or sequence of sections.

In some embodiments, the training of neural networks can be more computationally complex than dictionary learning for a similar accuracy, but the resulting model or algorithm can also be more flexible in representing visual data while using fewer coefficients for the reconstruction. In some embodiments, the resultant neural network model to be transmitted alongside the lower- quality visual data can be both smaller and can be more accurate in the reconstruction of the higher-quality visual data.

Some aspects can provide an improved technique for generating reconstruction parameters that can be used, when converting original high-quality visual data into a down-sampled low-quality visual data, to allow recreation of higher quality visual data without significant loss in quality, for example having a low reconstruction error in comparison with the original visual data, and with a reduction

in visual data transferred over a network. In such aspects, the application of such a technique can reduce the data transmitted when transmitting visual data in comparison with existing techniques while enabling reproduction of the visual data at its original quality without significant loss in quality in comparison to the original visual data (where quality can be defined by objective metrics such as error rate, PSNR and SSIM as well as subjective measures) or, alternatively, based on a perception measure or metric rather than on a pixel-wise comparison of images. In such aspects, such a proposed technique can allow minimal changes to be made to the overall infrastructure of service providers, as it can augment most existing compression techniques, and can provide advantages in encoding and streaming applications.

Optionally, the hierarchical algorithm can be used as a filter in the encoding or decoding of visual data.

In some embodiments, using the method as a filter for visual data codecs can provide very high computational efficiency, and therefore can also provide minimal energy costs in performing such filtering. In these or other embodiments, the method can provide a filter that is fast and/or flexible in expression and that can perform substantially accurate filtering in at least some embodiments.

Optionally, the higher-quality visual data is at a higher resolution than the lower-quality visual data, wherein the lower-quality visual data may contain a higher amount of artefacts than the higher-quality visual data.

In some embodiments, separating the visual data into a series of sections allows for the individual sections to be down-sampled thus reducing the visual data size, thereby allowing for lower quality sections to be transmitted as re-encoded visual data in the original or optionally a more optimal codec but at a lower resolution.

Optionally, the hierarchical algorithm performs image enhancement, preferably using super-resolution techniques. The hierarchical algorithm may also use a spatio-temporal approach.

In some embodiments, optionally for use for a section of visual data, the example based model may be a neural network and can use spatio-temporal convolution. In some embodiments, separating visual data into a series of sections allows for the individual sections to be down-sampled thus reducing the visual data size, thereby allowing for lower quality sections to be transmitted as re-encoded visual data in the original or optionally a more optimal codec but at a lower quality. In some embodiments, a spatio-temporal network can allow an improvement in performance by exploiting the temporal information in the visual data and, for example, within a similar scene in sequential sections of visual data, there may be stationary sections of background in the sequential sections providing information relevant for the higher-quality version of that scene such that temporally consecutive sections can be used to super resolve one section.

Optionally, enhancing the quality of visual data means upscaling the quality of the visual data. Furthermore, the plurality of input sections may comprise at least one low-quality input or a plurality of low-quality inputs, wherein quality can be measured subjectively.

Optionally, using the pre-processing or post-processing hierarchical algorithms may comprise any of: training the hierarchical algorithms; generating the hierarchical algorithms; or developing the hierarchical algorithms or applying the trained algorithm.

Aspects and/or embodiments include a computer program product comprising software code to effect the method and/or apparatus of other aspects and/or embodiments herein described.

It should be noted that in some aspects and/or embodiments, the terms model and/or algorithm and/or representation and/or parameters and/or functions can be used interchangeably.

It should also be noted that visual data, in some embodiments, may comprise image and/or video data.

References to visual data can be references to video data and/or image data in some aspects and/or embodiments and vice versa. References to low-quality and/or lower-quality can be references to low-resolution and/or lower-resolution in some aspects and/or embodiments and vice versa. References to high-quality and/or higher-quality and/or highest quality and/or original quality can be references to high-resolution and/or higher-resolution and/or highest-resolution and/or original resolution and/or increased fidelity in some aspects and/or embodiments and vice versa. References to sections can be references to frames and/or portions of frames in some aspects and/or embodiments and vice versa. References to enhance or enhancement can be references to upscale and/or upscaling in some aspects and/or embodiments and vice versa

Brief Description of Figures

Embodiments of the present invention will now be described, by way of example only with reference to the accompanying drawing, in which:

Figure 1a illustrates the layers in a convolutional neural network with no sparsity constraints;

Figure 1b illustrates the layers in a convolutional neural network with sparsity constraints; and

Figure 2 illustrates the method steps of pre-processing visual data frames prior to encoding for transmission, and decoding received data then post-processing to obtain decoded visual data frames.

Specific Description

With reference to Figures 1a and 1b, various possible configurations of neural network for use in at least some embodiments shall now be described in detail.

An example layered neural network is shown in Figure 1a having three layers 10, 20, 30, each layer 10, 20, 30 formed of a plurality of neurons 25, but where no sparsity constraints have been applied so all neurons 25 in each layer 10, 20, 30 are networked to all neurons 25 in any neighbouring layers 10, 20, 30. The example

simple neural network shown in Figure 2a is not computationally complex due to the small number of neurons 25 and layers. Due to the density of connections, however, the arrangement of the neural network shown in Figure 1a will not scale up easily to larger sizes of network, i.e. the connections between neurons/layers, easily as the computational complexity soon becomes too great as the size of the network increases which in some embodiments may be in a non-linear fashion.

Where neural networks need to be scaled up to work on inputs with a high number of dimensions, it can therefore become too computationally complex for all neurons 25 in each layer 10, 20, 30 to be networked to all neurons 25 in the one or more neighbouring layers 10, 20, 30. A predetermined initial sparsity condition is used to lower the computational complexity of the neural network, by limiting the number of connections between neurons and/or layers thus enabling a neural network approach to work with high dimensional data such as images.

An example of a neural network is shown in Figure 1b with sparsity constraints, according to at least one embodiment. The neural network shown in Figure 1b is arranged so that each neuron 25 is connected only to a small number of neurons 25 in the neighbouring layers 40, 50, 60 thus creating a neural network that is not fully connected and which can scale to function with, higher dimensional data – for example as an optimisation process for video. The smaller number of connections in comparison with a fully networked neural network allows for the number of connections between neurons to scale in a substantially linear fashion.

Alternatively, in some embodiments neural networks can be used that are fully connected or not fully connected but in different specific configurations to that described in relation to Figure 1b.

Further, in some embodiments, convolutional neural networks are used, which are neural networks that are not fully connected and therefore have less complexity than fully connected neural networks. Convolutional neural networks can also make use of pooling, for example max-pooling or mean-pooling, to reduce the dimensionality (and hence complexity) of the data that flows through the neural network and thus this can reduce the level of computation required. In some

embodiments, various approaches to reduce the computational complexity of convolutional neural networks can be used such as the Winograd algorithm or low-rank matrix approximations.

In one aspect, embodiments for a method and/or system for optimising an enhancement algorithm will now be described in detail with respect to Figure 2 as follows. The described embodiment relates to a method but in other embodiments can relate to a system and/or apparatus.

In method 200, a pre-processor step 220 and a post-processor step 260 are provided. In the described embodiment, visual data is input as visual data frames 210, such as video data, or sections of visual data can be used in other embodiments.

The pre-processor step 220 receives the visual data frames 210 and processes the visual data frames 210 using a trained processing algorithm. The trained processing algorithm used at the pre-processor step 222, is used to lossy encode the visual data 210. In some embodiments, the trained processing algorithm may be represented by $f_\phi : \mathbb{X} \rightarrow \mathbb{R}^N$ where ϕ are the parameters of a smooth encoder f. The pre-processor step 220 then outputs the pre-processed visual data to a standard encoder step 230.

The standard encoder step 230, which involves the use of a standard visual data codec, is paired with the standard decoding step 250, which involves the use of the same standard visual data codec. The standard encoder step 230 involves the creation of data packets 240, h, containing optimised video data that has been compressed. In some embodiments, the standard visual data codec may be represented as a function, h_θ, where θ are optional parameters. The standard visual data codec may be used to losslessly or lossyly encode the output of the pre-processor,$f_\phi(x)$, such that the data packets 240 are represented by:

$$z = h_\theta\left(f_\phi(x)\right)$$

Equation 1

where z is the bit representation assigned by the codec

In at least one embodiment, the standard visual data codec uses the H.264 standard but, in other embodiments, any other suitable video or image compression standard would be a suitable for use in the paired encoder and decoder steps 230, 250. At step 250, the decoder of the standard visual data codec, H_θ, processes the received data packets 240 such that:

$$\hat{z} = H_\theta\left(h_\theta\left(f_\phi(x)\right)\right)$$

Equation 2

Suitable alternative video or image compression standards include codecs that can be generalised to a differential approximation for training the algorithms used in the pre-processor step 220 and the post-processor step 260.

In some embodiments, at step 260, the decoded data packets, $\hat{z}$, that is the output of the decoding step 250, are reconstructed using a post processor represented by a further function, g_ψ, which is used to produce reconstructed the visual data, $\hat{x}$. The success of any reconstruction is quantified using a metric $d(x, \hat{x})$. The post-processor step 260 receives the decoded data packets from step 250, and produces decoded frames 270.

Therefore, in some embodiments, the reconstruction error of a particular algorithm may be represented by:

$$\mathcal{L}(\psi) = \mathbb{E}\left[d(x, g_\psi(H_\theta(z)))\right]$$

Equation 3

where the expectation, $\mathbb{E}$, is taken with respect to the data x and the corresponding bit representation z.

In at least one embodiment, the pre-processor step 220 and/or post-processor step 260 is implemented using a trained hierarchical algorithm or neural network,

such as a convolutional neural network, more details of which can be found in other embodiments in this specification and which can be applied in this aspect.

Training the hierarchical algorithms used in the pre-processor step 220 and/or post-processor step 260 is carried out as follows in at least one embodiment.

The pre-processor algorithms and the post-processor algorithms can be trained separately or together, but by training the pre-processor algorithms and the post-processor algorithms together a better output can be achieved in at least some embodiments. In some embodiments, the goal of training the pre-processor and post-processor algorithms is to obtain a small reconstruction error using as few bits as possible. That is, in some embodiments, the goal is to minimize:

$$\mathcal{L}(\theta,\phi,\psi) = \mathbb{E}\left[\left|h_\theta\left(f_\phi(x)\right)\right|\right] + \lambda\mathbb{E}\left[d\left(x, g_\psi\left(H_\theta\left(h_\theta\left(f_\phi(x)\right)\right)\right)\right)\right]$$

Equation 4

where the expectation is taken with respect to the data distribution, λ controls the trade-off between compression and reconstruction error and |h(f(x))| is the number of bits in h(f(x)). In some embodiments, H_θ and / or $h_\theta$cannot be differentiated, therefore a number of approximations may be made in order to minimise the number of bits, $\mathcal{L}(\theta,\phi,\psi)$, with respect to the parameters θ,ϕ,ψ or a subset of them.

First a differential approximation for the effects of the standard encoder and decoder is determined for use, depending on the codec used by the encoder 230 and the decoder 250. This differential approximation is based on the selected codec used by the encoder-decoder pair in each embodiment, for example H.264. A first-order approximation algorithm, which in this embodiment is a gradient descent algorithm (but other suitable algorithms can be used in other embodiments), is applied in the training process to determine a full end-to-end system encapsulating how visual data being provided to the encoder 230 will be decoded by the decoder 250.

Furthermore, in some embodiment, where it is necessary to optimise the parameters of the standard visual data codec in steps 230 and 25, approximate gradients may be calculated as follows:

$$\nabla_\theta \mathcal{L} \approx \nabla_\theta \mathbb{E}\left[n_\theta\left(f_\phi(x)\right)\right]$$

Equation 5

where n_θ represents a differential approximation of the number of bits used by the codec and may be used for optimizing the parameters of the codec.

$$\nabla_\phi \mathcal{L} \approx \nabla_\phi \mathbb{E}\left[n_\theta f_\phi(x)\right] + \lambda \nabla_\phi \mathbb{E}\left[d\left(x, g_\psi\left(q_\phi(f_\theta(x))\right)\right)\right]$$

Equation 6

where q_θ represents a differential approximation of the effects of the codec.

$$\nabla_\psi \mathcal{L} = \lambda \nabla_\psi \mathbb{E}\left[d\left(x, g_\psi\left(H_\theta\left(h_\theta\left(f_\phi(x)\right)\right)\right)\right)\right]$$

Equation 7

In some embodiments, *Equation 7* results in a minimization of an upper bound of the number of bits required.

Optionally, in some embodiments the generalised codec behaviour, such as the differential approximation of the codec, can be used as a middle (i.e. not the first or last) layer of a neural network (or hierarchical algorithm) and further optionally can be treated as a hidden layer that is fixed in functionality. The use of a differential approximation of the codec enables the use of techniques such as back-propagation so at to optimise the behaviour of the pre-encoder and post-decoder processes together in an end-to-end training framework. Such a layer effectively performs an encode process and then a decode process. Such a neural network/hierarchical algorithm can be trained where the input and output data is differentiable with respect to an object function, to allow for optimisation of the end-to-end process during training.

Optionally, in some embodiments, the neural network or hierarchical algorithm can be split or separated into three parts, where output layers have been enabled for each part, having (1) a pre-encoder layer; (2) the codec layer; and (3) the post-decoder layer.

Training can be performed on different source material as set out in other described aspects/embodiments, for example based on specific content or video samples/video.

In some embodiments, the training may enable certain pre-determined or hardware based criteria to be fixed so as to optimise the processing for specific hardware. For example, the ability to achieve the best optimisation for a particular bandwidth. Similarly, in some embodiments, quality criteria could be fixed so as to achieve the best possible optimisation which results in a particular quality.

Optimisation by the pre-processor step 220 allows for the input to the standard encoder 230 to be optimised, based on the trained pair of pre-processor and post-processor neural network/hierarchical algorithm for input into the post-processor step 260 by the standard decoder 250.

In some embodiments, the trained models may be used as part of a standard single image (or intra-image) codec, such as JPEG, or in other embodiments may be used as part of an intra-frame encoder in a video codec.

Any system feature as described herein may also be provided as a method feature, and vice versa.

As used herein, means plus function features may be expressed alternatively in terms of their corresponding structure.

In particular, method aspects may be applied to system aspects, and vice versa.

Furthermore, any, some and/or all features in one aspect can be applied to any, some and/or all features in any other aspect, in any appropriate combination.

It should also be appreciated that particular combinations of the various features described and defined in any aspects of the invention can be implemented and/or supplied and/or used independently.

In alternative embodiments, the input visual data concerned may be media for playback, such as recorded visual data or live streamed visual data, or it can be videoconference video or any other visual data source such as video recorded or being recorded on a portable device such as a mobile phone or a video recording device such as a video camera or surveillance camera.

It should also be appreciated that the term 'visual data', may refer to a single image, a sequence of images, video, or a section of visual data.

It should further be appreciated that the term "enhancing" may refer to upscaling, increasing the resolution and/or quality of visual data. References to enhancing or increasing the quality of visual data can refer to upscaling or using enhancement techniques of the possible embodiments described. References to down sampling can refer to reducing the resolution and/or quality of visual data (for example by quantisation to lower the bit rate of the visual data).

It should also be appreciated that the term 'frame', particularly in reference to grouping multiple frames into scenes, can refer to both an entire frame of a video and an area comprising a smaller section of a frame.

In aspects and/or embodiments, the terms algorithms and/or models and/or parameters can be used interchangeably or exchanged with each other. Further, in aspects and/or embodiments, the terms hierarchical algorithm, hierarchical model and hierarchical parameter can be exchanged with the terms convolutional neural networks and/or convolutional neural network model, convolutional neural network algorithm, convolutional neural network parameter.

Claims

1. A method for training a plurality of visual processing algorithms for processing visual data, the method comprising the steps of:

using a pre-processing hierarchical algorithm to process the visual data prior to encoding the visual data in visual data processing; and

using a post-processing hierarchical algorithm to further process the visual data following decoding visual data in visual data processing;

wherein the steps of encoding and decoding are performed with respect to a predetermined visual data codec.

2. The method according to claim 1 further comprising the step of receiving one or more sections of visual data.

3. The method of any previous claim, wherein processing the visual data comprises optimising the visual data.

4. The method of any previous claim, wherein one or more parameters associated with the pre-processing hierarchical algorithm may be stored in a library for re-use in encoding alternative visual data similar to the visual data used for training.

5. The method of any previous claim, wherein one or more parameters associated with the pre-processing hierarchical algorithm may be transmitted to a device configured to perform the method, for re-use in encoding alternative visual data similar to the visual data used for training, or with similar codec and encoder settings.

6. The method of any previous claim, wherein one or more parameters associated with the post-processing hierarchical algorithm, may be transmitted with any processed visual data to a remote device; wherein the

remote device has a pre-processing hierarchical algorithm associated with the post-processing hierarchical algorithm.

7. The method of any previous claim wherein the pre-processing and/or post-processing hierarchical algorithm comprises a layer that generalises the visual data processing.

8. The method according to any previous claim, wherein the pre-processing and/or post-processing hierarchical algorithm comprises a layer that generalises the encoding and/or decoding performed during visual data processing.

9. The method of any previous claim, wherein the method further comprises the step of receiving a plurality of criteria upon which to base any processing.

10. The method of claim 9, wherein the criteria is a specific bit rate.

11. The method of claim 9, wherein the criteria is a quality characteristic.

12. The method according to any previous claim, wherein the hierarchical algorithms comprise a plurality of connected layers.

13. The method according to claim 12, wherein the plurality of connected layers are any of sequential, recurrent, recursive, branching or merging.

14. The method according to any previous claim where the visual data comprises one or more sections of visual data.

15. The method according to any previous claim, wherein visual data comprises any of: a single frame of visual data, a sequence of frames of visual data, and a region within a frame or sequence of frames of visual data.

16. The method according to any previous claim, wherein the visual data comprises a plurality of frames of video, or a plurality of images.

17. The method according to claim 14 or any preceding claim when dependent on claim 14, wherein the hierarchical algorithm differs for each section of visual data.

18. The method according to any previous claim, wherein the hierarchical algorithm is selected from a library of algorithms.

19. The method according to claim 18, wherein standardised features of the at least one section of received lower-quality visual data are extracted and used to select the hierarchical algorithm from the library of algorithms.

20. The method according to any previous claim, wherein the hierarchical algorithm to be selected from the library of algorithms is based on generating the highest quality version of the lower-quality visual data, preferably wherein quality can be defined by any of: an error rate; a bit error rate; a peak signal-to-noise ratio; or a structural similarity index.

21. The method according to any previous claim, wherein the hierarchical algorithms are developed using a learned approach.

22. The method according to claim 21, wherein the learned approach comprises machine learning techniques.

23. The method according to any previous claim, wherein the hierarchical algorithm is a non-linear hierarchical algorithm.

24. The method according to claim 23, wherein the non-linear hierarchical algorithm comprises one or more convolutional neural networks.

25. The method according to any previous claim, wherein the hierarchical algorithm can be used as a filter in the encoding or decoding of visual data.

26. The method according to any previous claim, wherein the higher-quality visual data is at a higher resolution than the lower-quality visual data.

27. The method according to any previous claim, wherein the lower-quality visual data contains a higher amount of artefacts than the higher-quality visual data.

28. The method according to any previous claim, wherein the hierarchical algorithm performs image enhancement, preferably using super-resolution techniques.

29. The method according to any previous claim, wherein the hierarchical algorithm uses a spatio-temporal approach.

30. The method of any previous claim, wherein enhancing the quality of visual data means upscaling the quality of the visual data.

31. The method of any previous claim, wherein using the pre-processing or post-processing hierarchical algorithms comprises any of: training the hierarchical algorithms; generating the hierarchical algorithms; or developing the hierarchical algorithms.

32. The method of any previous claim, wherein visual data processing comprises a codec which compresses the visual data into a data representation.

33. The method of claim 32 wherein the data representation is smaller than the visual data.

34. A method substantially as hereinbefore described in relation to Figures 1 and 2.

35. Apparatus for carrying out the method of any preceding claim.

36. A computer program product comprising software code for carrying out the method of any of claims 1 to 34.

Figure 1b

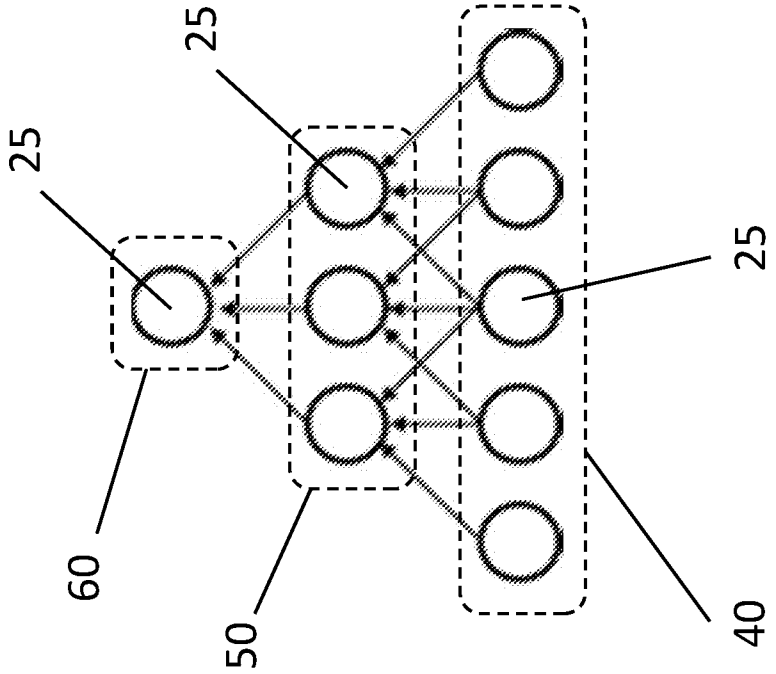
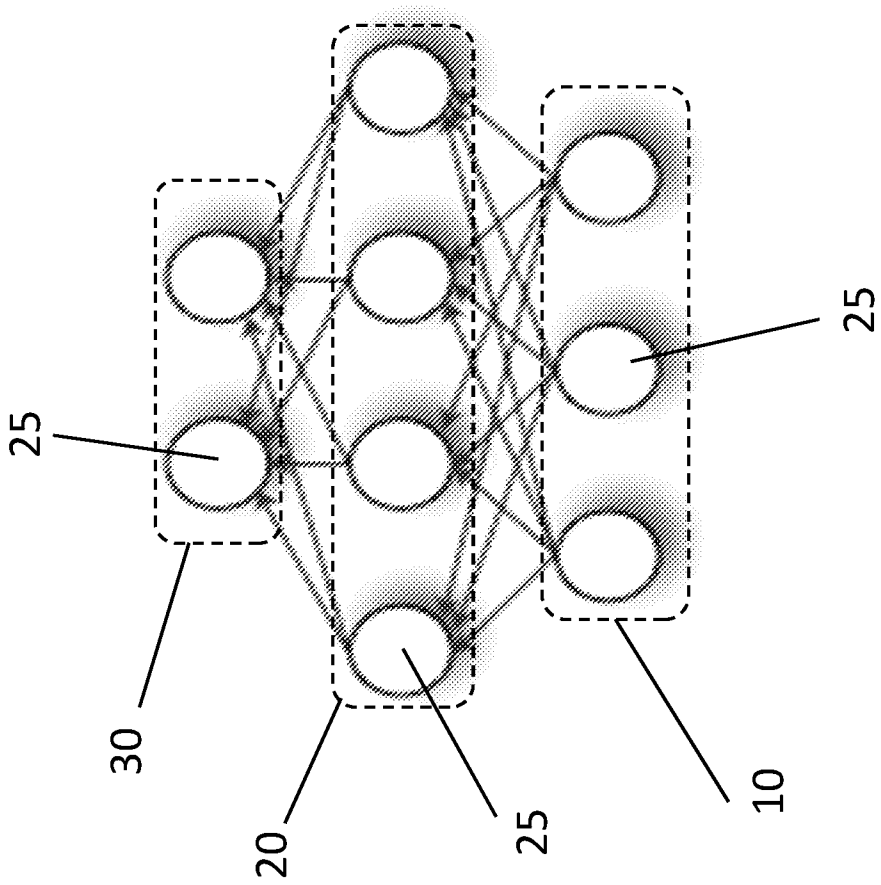


Figure 1a



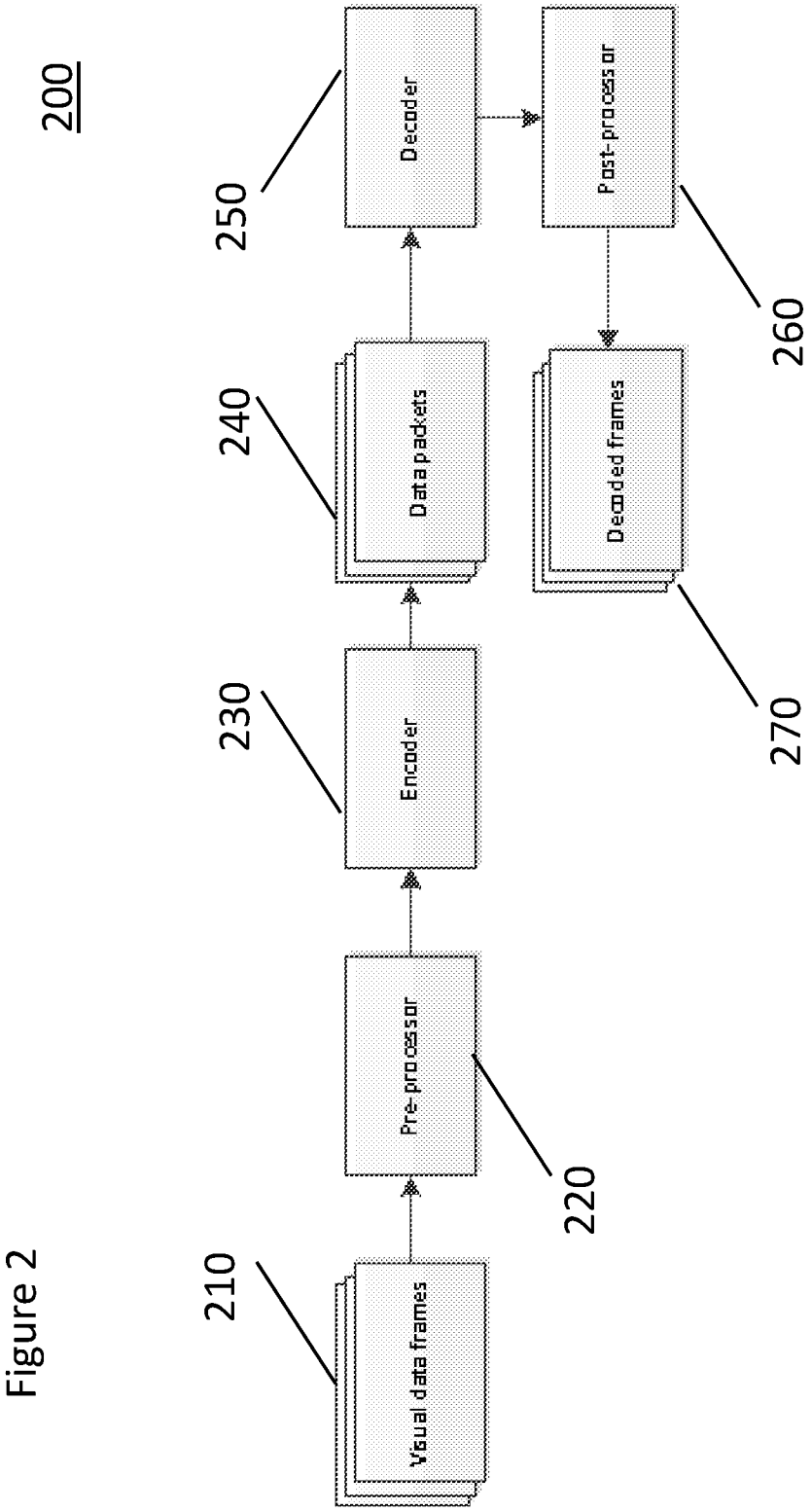

200
Visual data frames
Pre-processor
Encoder
Data packets
Decoder
Post-processor
Decoded frames
210
220
230
240
250
260
270

Figure 2

INTERNATIONAL SEARCH REPORT

International application No
PCT/GB2016/050922

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N19/46 G06T3/40 G06T5/00 H04N19/117 H04N19/142
H04N19/154 H04N19/177 H04N19/80 H04N19/86 H04N19/59
H04N19/147 H04N19/172 H04N19/136 H04N19/19 H04N19/85
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N G06T

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2004/086039 A1 (REYNOLDS JODIE L [US] ET AL) 6 May 2004 (2004-05-06) paragraph [0002] - paragraph [0008]; figures 2-8,11 paragraph [0026] - paragraph [0036] paragraph [0041] - paragraph [0067] -----	1-36
X	WO 2008/133951 A2 (MASSACHUSETTS INST TECHNOLOGY [US]; SEUNG H SEBASTIAN [US]; MURRAY JOS) 6 November 2008 (2008-11-06) abstract page 1, line 6 - page 2, line 4 page 7, line 18 - page 12, line 23 page 15, line 30 - page 17, line 6 ----- -/--	1-36

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 June 2016	13/06/2016

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Le Guen, Benjamin

1

INTERNATIONAL SEARCH REPORT

International application No
PCT/GB2016/050922

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	DONG CHAO ET AL: "Learning a Deep Convolutional Network for Image Super-Resolution", 6 September 2014 (2014-09-06), CORRECT SYSTEM DESIGN; [LECTURE NOTES IN COMPUTER SCIENCE; LECT.NOTES COMPUTER], SPRINGER INTERNATIONAL PUBLISHING, CHAM, PAGE(S) 184 - 199, XP047296566, ISSN: 0302-9743 ISBN: 978-3-642-27584-5 the whole document -----	1-36

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/GB2016/050922

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004086039 A1	06-05-2004	AU 2003285058 A1	08-06-2005
		CA 2542800 A1	02-06-2005
		EP 1680918 A1	19-07-2006
		JP 4463765 B2	19-05-2010
		JP 2007529125 A	18-10-2007
		US 2004086039 A1	06-05-2004
		US 2009310671 A1	17-12-2009
		WO 2005050988 A1	02-06-2005
WO 2008133951 A2	06-11-2008	US 2010183217 A1	22-07-2010
		WO 2008133951 A2	06-11-2008