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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
05 January 2023 (05.01.2023)

WIPO | PCT

(10) International Publication Number
WO 2023/274392 A1

(51) International Patent Classification:
H04N 19/59 (2014.01) *G06N 3/04* (2006.01)
G06T 3/40 (2006.01)

(21) International Application Number:
PCT/CN2022/103238

(22) International Filing Date:
01 July 2022 (01.07.2022)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
PCT/CN2021/104098
01 July 2021 (01.07.2021) CN

(71) Applicants: **BEIJING BYTEDANCE NETWORK TECHNOLOGY CO., LTD.** [CN/CN]; Room B-0035, 2/F, No.3 Building, No.30, Shixing Road, Shijingshan District, Beijing 100041 (CN). **BYTEDANCE INC.** [US/US]; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US).

(72) Inventors: **LIN, Chaoyi**; Jinritoutiao Post Office, China Satellite Communications Tower, No.63, Zhichun Road, Haidian District, Beijing 100080 (CN). **LI, Yue**; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US). **ZHANG, Kai**; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US). **ZHANG, Zhaobin**; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US). **ZHANG, Li**; 12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137, Los Angeles, California 90066 (US).

(74) Agent: **LIU, SHEN & ASSOCIATES**; 10th Floor, Building 1, 10 Caihefang Road, Haidian District, Beijing 100080 (CN).

(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, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, 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, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM,

(54) **Title:** Utilizing Coded Information During Super Resolution Process

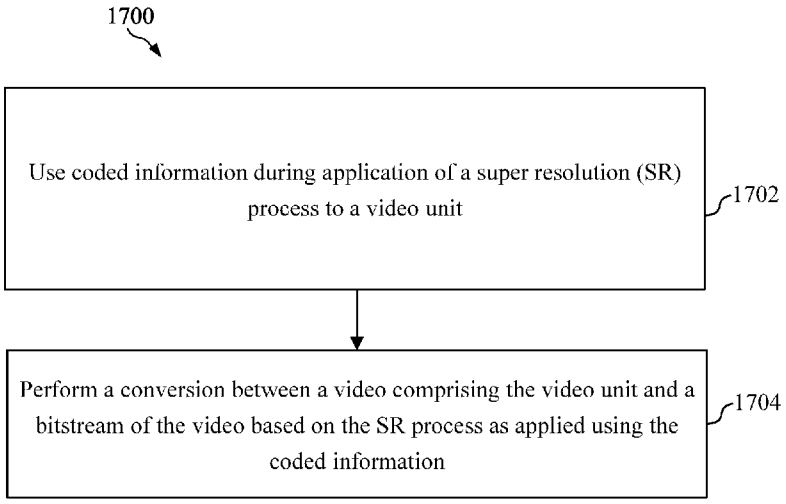


FIG. 17

(57) **Abstract:** A method of processing video data. The method includes using coded information during application of a super resolution (SR) process to a video unit; and performing a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information. A corresponding video coding apparatus and non-transitory computer-readable recording medium are also disclosed.

WO 2023/274392 A1

[Continued on next page]

TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, 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).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

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

Utilizing Coded Information During Super Resolution Process

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This patent application is a continuation of International Application No. PCT/CN2021/104098 filed July 1, 2021, by Beijing ByteDance Network Technology Co., Ltd., et al., and titled "On Convolutional Neural Network And Inputs For Super-Resolution-Based Video Coding," which is hereby incorporated by reference.

TECHNICAL FIELD

[0002] The present disclosure is generally related to video coding and, in particular, to super-resolution-based up-sampling for video coding.

BACKGROUND

[0003] Digital video accounts for the largest bandwidth use on the internet and other digital communication networks. As the number of connected user devices capable of receiving and displaying video increases, it is expected that the bandwidth demand for digital video usage will continue to grow.

SUMMARY

[0004] The disclosed aspects/embodiments provide techniques that used coded information (e.g., coded information/decoded information) during a super resolution (SR) process. For example, the coded information may be partition information, prediction information, or an intra prediction mode. The coded information may be used as an input to a neural network (NN) of the SR process. The techniques also provide techniques whereby color components (e.g., luma samples and/or chroma samples) are used as an input during the SR process. The techniques can be used by video and image encoding, decoding, streaming and storing implementations. Thus, the video coding process is improved relative to conventional video coding techniques.

[0005] A first aspect relates to a method of processing video data. The method includes using coded information during application of a super resolution (SR) process to a video unit; and performing a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information.

[0006] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises encoded information.

[0007] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises decoded information.

[0008] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the SR process comprises a neural network (NN)-based SR process, and wherein the coded information is used as an input to the NN-based SR process.

[0009] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information is used to determine that the SR process, which is one of a plurality of SR processes available, is applied.

[0010] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises one or more of partition information, prediction information, and an intra prediction mode.

[0011] Optionally, in any of the preceding aspects, another implementation of the aspect provides that an input of the SR process comprises reconstructed low-resolution samples and decoded information, and wherein the decoded information comprises one or more of partition information, prediction information, and an intra prediction mode.

[0012] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises partition information having a same resolution as a reconstructed low-resolution frame.

[0013] Optionally, in any of the preceding aspects, another implementation of the aspect provides deriving sample values in the partition information by averaging reconstructed samples in the video unit.

[0014] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises prediction information, and wherein the prediction information comprises prediction samples from intra prediction, intra block copy (IBC), or inter-prediction.

[0015] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises an intra prediction mode having a same resolution as a reconstructed low-resolution frame.

[0016] Optionally, in any of the preceding aspects, another implementation of the aspect provides deriving sample values in the intra prediction mode by filling the intra prediction mode in a corresponding video unit.

[0017] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises quantization parameter (QP) information, and wherein the QP information is used to generate an up-sampled reconstruction.

[0018] Optionally, in any of the preceding aspects, another implementation of the aspect provides filling a matrix with quantization (QP) parameter values to construct a QP map, wherein the QP map is used as an input to the SR process, and wherein a spatial resolution of the QP map is the same as the spatial resolution of another input of the SR process.

[0019] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the video unit comprises a first color component, and wherein a second color component is utilized during the SR process applied to the first color component.

[0020] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the video unit comprises a first color component, and wherein a second color component is used as an input to the SR process applied to the first color component.

[0021] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the SR process comprises a luma up-sampling process, and wherein chroma information is used as an input to the luma up-sampling process.

[0022] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the SR process comprises a chroma up-sampling process, and wherein luma information is used as an input to the chroma up-sampling process.

[0023] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma information comprises luma samples of reconstruction, and wherein the luma samples of reconstruction are used as an input to the SR process after in-loop filters have been applied to the video unit or before the in-loop filters have been applied to the video unit.

[0024] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma information comprises luma samples of reconstruction, and wherein the luma samples of reconstruction are used as an input to the SR process after a first one of a plurality of in-loop filters have been applied to the video unit but before a last one of the plurality of in-loop filters have been applied to the video unit.

[0025] Optionally, in any of the preceding aspects, another implementation of the aspect provides that an input to a neural network (NN) in the SR process includes both chroma samples of reconstruction and luma samples of reconstruction.

[0026] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma samples of reconstruction are down-sampled to a same resolution as the chroma samples of reconstruction and are concatenated with the chroma samples of reconstruction.

[0027] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma samples of reconstruction are down-sampled using bi-linear interpolation.

[0028] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma samples of reconstruction are down-sampled using bi-cubic interpolation.

[0029] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma samples of reconstruction are down-sampled using convolution with a stride equal to a scaling ratio of an original frame.

[0030] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma samples of reconstruction are down-sampled using an inverse of a pixel shuffle.

[0031] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the luma samples of reconstruction are down-sampled based on a color format of 4:2:0 or 4:2:2.

[0032] Optionally, in any of the preceding aspects, another implementation of the aspect provides that a method of down-sampling the luma samples of reconstruction is signaled from an encoder to a decoder.

[0033] Optionally, in any of the preceding aspects, another implementation of the aspect provides that a determination to apply down-sampling to the luma samples of reconstruction is made based on a color format.

[0034] Optionally, in any of the preceding aspects, another implementation of the aspect provides that a color format is 4:4:4 and the luma samples of reconstruction are not down-sampled.

[0035] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the chroma information comprises chroma samples of reconstruction, and wherein the chroma samples of reconstruction are used as an input to the SR process after in-loop filters have been applied to the video unit or before the in-loop filters have been applied to the video unit.

[0036] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the chroma information comprises chroma samples of reconstruction, and wherein the chroma samples of reconstruction are used as an input to the SR process after a first one of a plurality of in-loop filters have been applied to the video unit but before a last one of the plurality of in-loop filters have been applied to the video unit.

[0037] Optionally, in any of the preceding aspects, another implementation of the aspect provides that an input to a neural network (NN) in the SR process includes both chroma samples of reconstruction and luma prediction samples.

[0038] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the first color component is a chroma component and the second color component is a different chroma component.

[0039] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the input includes reconstructed samples and decoded information, and wherein the decoded information comprises one or more of partition information, prediction information, and an intra prediction mode.

[0040] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the intra prediction mode comprises a binary frame, wherein each value in the binary frame indicates whether or not one of the reconstructed samples belongs to a skip mode.

[0041] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the prediction information is derived by motion compensation for an inter predicted coding unit.

[0042] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises prediction information, and wherein the prediction information is used as an input of the SR process applied to the video unit for reconstruction.

[0043] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the prediction information comprises luma information of prediction pictures, and wherein the video unit comprises a luma component.

[0044] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the prediction information comprises luma information of prediction pictures, and wherein the video unit comprises a chroma component.

[0045] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the prediction information comprises chroma information of prediction pictures, and wherein the video unit comprises a chroma component.

[0046] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the prediction information comprises luma and chroma information of prediction pictures used together for reconstruction, and wherein the video unit comprises one or more of a luma component and a chroma component.

[0047] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the prediction information comprises padded samples when a coding mode is a palette mode or a pulse-code modulation mode.

[0048] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises partition information, and wherein the partition information is used as an input of the SR process applied to the video unit for reconstruction.

[0049] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the video unit comprises a low-resolution frame, and wherein the partition information has a same resolution as the low-resolution frame.

[0050] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the coded information comprises an intra prediction mode, and wherein the intra prediction mode is used as an input of the SR process applied to the video unit for reconstruction.

[0051] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the intra prediction mode is an intra prediction mode of a current sample obtained by intra prediction or inter prediction.

[0052] Optionally, in any of the preceding aspects, another implementation of the aspect provides that a matrix of the intra prediction mode is constructed as one input of the SR process, wherein the matrix has a same resolution as the video unit when reconstructed, and wherein each sample in the matrix has a value corresponding to the intra prediction mode of a corresponding coding unit (CU).

[0053] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the video unit comprises an intra coded slice (I slice) or an intra coded frame (I frame).

[0054] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the video unit comprises a uni-directional coded slice (P slice) or a bi-directional coded frame (B frame).

[0055] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the conversion includes encoding the video data into the bitstream.

[0056] Optionally, in any of the preceding aspects, another implementation of the aspect provides that the conversion includes decoding the video data from the bitstream.

[0057] An apparatus for processing media data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to: use coded information during application of a super resolution (SR)

process to a video unit; and perform a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information.

[0058] A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises: using coded information during application of a super resolution (SR) process to a video unit; and generating the bitstream based on the SR process as applied using the coded information.

[0059] An apparatus for processing media data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to perform a method recited in any of the disclosed embodiments.

[0060] A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method, recited in any of the disclosed embodiments, performed by a video processing apparatus.

[0061] A computer-readable program medium having code stored thereupon, the code comprising instructions that, when executed by a processor, causes the processor to implement a method recited in any of the disclosed embodiments.

[0062] For the purpose of clarity, any one of the foregoing embodiments may be combined with any one or more of the other foregoing embodiments to create a new embodiment within the scope of the present disclosure.

[0063] These and other features will be more clearly understood from the following detailed description taken in conjunction with the accompanying drawings and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0064] For a more complete understanding of this disclosure, reference is now made to the following brief description, taken in connection with the accompanying drawings and detailed description, wherein like reference numerals represent like parts.

[0065] FIG. 1 is a schematic diagram illustrating an example application of reference picture resampling (RPR).

[0066] FIG. 2 is a schematic diagram illustrating an example of de-convolution.

[0067] FIG. 3 is a schematic diagram illustrating an example of a process of pixel shuffle-based up-sampling.

[0068] FIG. 4 is a schematic diagram of an example for an up-sampling network.

[0069] FIG. 5 is a schematic diagram illustrating an example of obtaining residual blocks, where M denotes the number of filters.

[0070] FIGS. 6A-6D are schematic diagrams illustrating examples of different positions for up-sampling.

[0071] FIG. 7 is a schematic diagram of an example of an inverse process of the pixel shuffle.

[0072] FIG. 8 is a schematic diagram of an overall framework for up-sampling according to an embodiment of the disclosure.

[0073] FIG. 9 is a schematic diagram of an example of a neural network (NN) for reconstruction of a Y channel.

[0074] FIG. 10 is a schematic diagram of an example of a pixel shuffle operator.

[0075] FIG. 11 is a schematic diagram of an example of a neural network for reconstruction of U and V channels.

[0076] FIG. 12 is a block diagram showing an example video processing system.

[0077] FIG. 13 is a block diagram of a video processing apparatus.

[0078] FIG. 14 is a block diagram that illustrates an example of a video coding system.

[0079] FIG. 15 is a block diagram illustrating an example of a video encoder.

[0080] FIG. 16 is a block diagram illustrating an example of a video decoder.

[0081] FIG. 17 is a method of processing video data according to an embodiment of the disclosure.

DETAILED DESCRIPTION

[0082] It should be understood at the outset that although an illustrative implementation of one or more embodiments are provided below, the disclosed systems and/or methods may be implemented using any number of techniques, whether currently known or in existence. The disclosure should in no way be limited to the illustrative implementations, drawings, and techniques illustrated below, including the exemplary designs and implementations illustrated and described herein, but may be modified within the scope of the appended claims along with their full scope of equivalents.

[0083] Video coding standards have evolved primarily through the development of the well-known International Telecommunication Union – Telecommunication (ITU-T) and International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) standards. The ITU-T produced H.261 and H.263, ISO/IEC produced Moving Picture Experts Group (MPEG)-1 and MPEG-4 Visual, and the two organizations jointly produced the H.262/MPEG-2 Video and H.264/MPEG-4 Advanced Video Coding (AVC) and H.265/High Efficiency Video Coding (HEVC) standards.

[0084] Since H.262, the video coding standards are based on the hybrid video coding structure wherein temporal prediction plus transform coding are utilized. To explore the future video coding technologies beyond HEVC, Joint Video Exploration Team (JVET) was founded by Video Coding Experts Group (VCEG) and MPEG jointly in 2015. Since then, many new methods have been adopted by JVET and put into the reference software named Joint Exploration Model (JEM).

[0085] In April 2018, the Joint Video Expert Team (JVET) between VCEG (Q6/16) and ISO/IEC JTC1 SC29/WG11 (MPEG) was created to work on the Versatile Video Coding (VVC) standard targeting at fifty percent (50%) bitrate reduction compared to HEVC. VVC version 1 was finalized in July 2020.

[0086] The latest version of VVC, which is known as H.266, is embodied in the ITU-T document entitled "Versatile Video Coding," published August 2020. The reference software for VVC is known as the VVC Test Model (VTM). The VTM is embodied in the JVET document entitled "JVET-Software Manual," by Bossen, et al., published August 13, 2020. H.266 terminology is used in some description only for ease of understanding and not for limiting scope of the disclosed techniques. As such, the techniques described herein are applicable to other video codec protocols and designs also.

[0087] FIG. 1 is a schematic diagram illustrating an example application of reference picture resampling (RPR) 100. RPR is a new mechanism in VVC where pictures in the reference lists can be stored at a different resolution from the current picture and then resampled in order to perform regular decoding operations. The inclusion of this technique supports interesting application scenarios such as real-time communication with adaptive resolution, adaptive streaming with open group of pictures (GOP) structures. As shown in FIG. 1, a down-sampled (a.k.a., downsampled, or down sampled) sequence is encoded and then the reconstruction is up-sampled (a.k.a., upsampled, or up sampled) after decoding.

[0088] Commonly used or traditional up-sampling technology is discussed. In VTM 11.0, the up-sampling filter is a discrete cosine transform (DCT)-Based Interpolation Filter (DCTIF). Besides that, bi-cubic interpolation and bi-linear interpolation are also commonly used. In these technologies, the weight coefficients for the interpolation filter are fixed once the number of taps of filters is given. Thus, the weight coefficients of these methods may be not the optimal.

[0089] FIG. 2 is a schematic diagram illustrating an example of de-convolution 200. De-convolution, which is also referred to as transposed convolution, is usually used for up-sampling in deep learning. In this method, the stride for convolution is the same as the scaling ratio. The bottom matrix is the low-resolution input where white blocks are the padded value with zeros

and the gray blocks denote the original samples in low-resolution. The top matrix is the high-resolution output. In this example, the stride=2.

[0090] FIG. 3 is a schematic diagram illustrating an example of a process of pixel shuffle-based up-sampling 300. The pixel shuffle layer is described in W. Shi, J. Caballero, et al., "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network," Proceedings of the IEEE conference on computer vision and pattern recognition, 2016. The pixel shuffle layer is another method for up-sampling used in deep learning. As shown in FIG. 3, the pixel shuffle layer is usually placed after a convolutional layer. The number of filters of this convolution is M = $C_{out}r^2$, where C_{out} is the number of output channels and r denotes the up-scaling ratio. For example, given a low-resolution input with the size of H×W×3, when the size of high-resolution output is 2H×2W×3, then the number of filters M = 3×2^2=12. The pixel shuffle technique is described in further detail with regard to FIGS. 7 and 10, below.

[0091] Convolutional neural network-based super resolution for video coding is discussed. Super-resolution (SR) is the process of recovering high-resolution (HR) images from low-resolution (LR) images. SR may also be referred to as up-sampling. In deep learning, a convolutional neural network (a.k.a., CNN or ConvNet) is a class of deep neural networks commonly applied to analyzing visual imagery. CNNs have very successful applications in image and video recognition/processing, recommender systems, image classification, medical image analysis, and natural language processing.

[0092] CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "fully-connectedness" of these networks makes them prone to overfitting data. Typical ways of regularization include adding some form of magnitude measurement of weights to the loss function. CNNs take a different approach towards regularization. That is, CNNs take advantage of the hierarchical pattern in data and assemble more complex patterns using smaller and simpler patterns. Therefore, on the scale of connectedness and complexity, CNNs are on the lower extreme.

[0093] CNNs use relatively little pre-processing compared to other image classification/processing algorithms. This means that the network learns the filters that in traditional algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major advantage.

[0094] Deep learning for image/video coding is discussed. Deep learning-based image/video compression typically has two implications: end-to-end compression purely based on neural

networks (NNs) and traditional frameworks enhanced by neural networks. The first type usually takes an auto-encoder like structure, either achieved by convolutional neural networks or recurrent neural networks. While purely relying on neural networks for image/video compression can avoid any manual optimizations or hand-crafted designs, compression efficiency may be not satisfactory. Therefore, works distributed in the second type take neural networks as an auxiliary, and enhance traditional compression frameworks by replacing or enhancing some modules. In this way, they can inherit the merits of the highly optimized traditional frameworks.

[0095] CNN-based super resolution is discussed in further detail. In lossy image/video compression, the reconstructed frame is an approximation of the original frame, since the quantization process is not invertible and thus incurs distortion to the reconstructed frame. In the context of RPR, the input image/video may be down-sampled. Thus, the resolution of original frame is 2x of that of reconstruction. To up-sample the low-resolution reconstruction, a convolutional neural network could be trained to learn the mapping from the distorted low-resolution frame to the original high-resolution frame. In practice, training must be performed prior to deploying the NN-based in-loop filtering. For example, see the CNN-based block up-sampling method for HEVC proposed in J. Lin, et al., "Convolutional Neural Network-Based Block Up-Sampling for HEVC", TCSVT 2019. For each coding tree unit (CTU) block, the method determines whether to use a down/up-sampling based method or the full-resolution based coding.

[0096] Training is discussed. The purpose of the training processing is to find the optimal value of parameters including weights and bias. First, a codec (e.g., the HEVC test model (HM), Joint Exploration Model (JEM), VTM, etc.) is used to compress the training dataset to generate the distorted reconstruction frames.

[0097] The reconstructed frames (low-resolution and compressed) are then fed into the NN and the cost is calculated using the output of NN and the ground-truth frames (a.k.a., original frames). Commonly used cost functions include Sum of Absolution Difference (SAD) and Mean Square Error (MSE). Next, the gradient of the cost with respect to each parameter is derived through the back propagation algorithm. With the gradients, the values of the parameters can be updated. The above process repeats until the convergence criteria is met. After completing the training, the derived optimal parameters are saved for use in the inference stage.

[0098] The convolution process is discussed. During convolution, the filter is moved across the image from left to right, top to bottom, with a one-pixel column change on the horizontal movements, then a one-pixel row change on the vertical movements. The amount of movement

between applications of the filter to the input image is referred to as the stride. The stride is almost always symmetrical in height and width dimensions. The default stride or strides in two dimensions is (1,1) for the height and the width movement.

[0099] In most of deep convolutional neural networks, residual blocks are utilized as the basic module and stacked several times to construct the final network. FIG. 5 is a schematic diagram illustrating an example of obtaining residual blocks 500, where M denotes the number of filters. As shown in the example of FIG. 5, the residual block is obtained by combining a convolutional layer, a rectified linear unit (ReLU)/parametric rectified linear unit (PReLU) activation function, and a convolutional layer as shown in FIG. 5.

[00100] Interference is discussed. During the inference stage, the distorted reconstruction frames are fed into an NN and processed by the NN model whose parameters are already determined in the training stage. The input samples to the NN can be reconstructed samples before or after deblocking (DB), or reconstructed samples before or after sample adaptive offset (SAO), or reconstructed samples before or after adaptive loop filter (ALF).

[00101] Unfortunately, the existing NN-based super resolution for video coding has problems or drawbacks. First, NN-based super resolution does not make full use of available information, e.g., prediction, partition, intra prediction mode, and other modes, while this information may be beneficial to the filtering. Second, given a sequence to be compressed, existing methods perform the up-sampling either at the frame-level or block-level. However, up-sampling either at the frame-level or block-level is not always optimal due to the varied content of natural videos. For example, some frames might benefit from frame-level up-sampling while other frames might benefit from block-level up-sampling. Thus, it is better to exploit combining these two levels together.

[00102] Disclosed herein are techniques that used coded information (e.g., coded information/decoded information) during a super resolution (SR) process. For example, the coded information may be partition information, prediction information, or an intra prediction mode. The coded information may be used as an input to a neural network (NN) of the SR process. The techniques also provide techniques whereby color components (e.g., luma samples and/or chroma samples) are used as an input during the SR process. The techniques can be used by video and image encoding, decoding, streaming and storing implementations. Thus, the video coding process is improved relative to conventional video coding techniques.

[00103] To solve the above problems and some other problems not mentioned, methods as summarized below are disclosed. The detailed embodiments below should be considered as

examples to explain general concepts. These embodiments should not be interpreted in a narrow way. Furthermore, these embodiments can be applied individually or combined in any manner.

[00104] In the present disclosure, an NN-based SR can be any kind of NN-based methods, such as a convolutional neural network (CNN) based SR. In the following discussion, an NN-based SR may also be referred to as a non-CNN-based method, e.g., using machine learning based solutions.

[00105] In the following discussion, a video unit (a.k.a., video data unit) may be a sequence of pictures, a picture, a slice, a tile, a brick, a subpicture, a CTU/coding tree block (CTB), a CTU/CTB row, one or multiple coding units (CUs)/coding blocks (CBs), one or multiple CTUs/CTBs, one or multiple Virtual Pipeline Data Unit (VPDU), or a sub-region within a picture/slice/tile/brick. In some embodiments, the video unit may be referred to as a video data unit.

[00106] **Example 1**

[00107] 1. It is proposed that the coded (encoded/decoded) information can be utilized during the super resolution process.

[00108] a. In one example, the coded information could be used as inputs to NN-based SR solutions.

[00109] b. In one example, the coded information could be used to determine which SR solution to be applied.

[00110] c. In one example, the coded information may include the partition information, the prediction information, and the intra prediction mode, etc.

[00111] i. In one example, the input includes the reconstructed low-resolution samples and other decoded information (e.g., the partition information, the prediction information, and the intra prediction mode).

[00112] ii. In one example, the partition information has the same resolution as the reconstructed low-resolution frame. Sample values in the partition are derived by averaging the reconstructed samples in a coding unit.

[00113] iii. In one example, the prediction information may be the generated prediction samples from intra prediction or IBC prediction or inter-prediction.

[00114] iv. In one example, the intra prediction mode has the same resolution as the reconstructed low-resolution frame. Sample values in the intra prediction mode are derived by filling the intra prediction mode in the corresponding coding unit.

[00115] v. In one example, the QP value information can be used as assistant information to improve the quality of up-sampled reconstruction.

[00116] 1. In one example, construct a QP map by filling a matrix with QP value and its spatial size is the same with other input data. The QP map will be fed into the network of super resolution.

[00117] **Example 2**

[00118] This example involves the color components for input of the SR network.

[00119] 2. Information related to a first color component may be utilized during the SR process applied to a second color component.

[00120] a. Information related to a first color component may be utilized as input for the SR process applied to a second color component.

[00121] b. Chroma information may be utilized as input for luma up-sampling process.

[00122] c. Luma information may be utilized as input for chroma up-sampling process.

[00123] i. In one example, the luma reconstructed samples before the in-loop filters may be used.

[00124] 1. Alternatively, the luma reconstructed samples after the in-loop filters may be used.

[00125] ii. In one example, the input to the NN contains both chroma reconstructed samples and luma reconstructed samples.

[00126] 1. In one example, the luma information can be down sampled to the same resolution with chroma components. The down-sampled luma information will be concatenated with the chroma components.

[00127] a. In one example, the down-sample method is bi-linear interpolation.

[00128] b. In one example, the down-sample method is bi-cubic interpolation.

[00129] c. In one example, the down-sample method is convolution with stride equal to the scaling ratio for original frame.

[00130] d. In one example, the down-sample method is the inverse of pixel shuffle, which is illustrated in FIG. 7. A high-resolution block (HR block) with size 4x4x1 will be down-sampled to a low-resolution block (LR block) with size 2x2x4. The font of first element in each channel of the LR block and the corresponding position in the HR block are bold.

[00131] e. In one example, the down-sample method may depend on color format such as 4:2:0 or 4:2:2.

[00132] f. In one example, the down-sample method may be signaled from the encoder to the decoder.

[00133] g. Alternatively, furthermore, whether to apply the down-sample process may depend on the color format.

[00134] 2. In another example, the color format is 4:4:4 and no down-sampling is performed to the luma information.

[00135] iii. In one example, the chroma reconstructed samples before the in-loop filters may be used.

[00136] 1. Alternatively, the chroma reconstructed samples after the in-loop filters may be used.

[00137] iv. In one example, the input to the NN contains both chroma reconstructed samples and luma reconstructed samples.

[00138] v. In one example, the input to the NN contains both chroma reconstructed samples and luma prediction samples.

[00139] d. In one example, one chroma component (e.g., Cb) information may be utilized as input for the other chroma component (e.g., Cr) up-sampling process.

[00140] e. In one example, the input includes the reconstructed samples and the decoded information (e.g., the mode information, and the prediction information).

[00141] i. In one example, the mode information is a binary frame with each value indicating if the sample belongs to a skip coded unit or not.

[00142] ii. In one example, the prediction information is derived via the motion compensation for inter coded coding unit.

[00143] 3. In one example, the prediction information may be utilized as input for the SR process applied to the reconstruction.

[00144] a. In one example, the luma information of prediction pictures may be utilized as input for the SR process of the luma component of reconstructions.

[00145] b. In one example, the luma information of prediction pictures may be utilized as input for the SR process of the chroma component of reconstructions.

[00146] c. In one example, the chroma information of prediction pictures may be utilized as input for the SR process of the chroma component of reconstructions.

[00147] d. In one example, the luma and chroma information of prediction pictures may be utilized together as input for the SR process of the reconstruction (for example, luma reconstruction).

[00148] e. In case prediction information is unavailable (such as the coding mode is palette or PCM), the prediction samples are padded.

[00149] 4. In one example, the partition information may be utilized as input for the SR process applied to the reconstruction.

[00150] a. In one example, the partition information has the same resolution as the reconstructed low-resolution frame. Sample values in the partition are derived by averaging the reconstructed samples in a coding unit.

[00151] 5. In one example, the intra prediction mode information may be utilized as input for the SR process applied to the reconstruction.

[00152] a. In one example, the intra prediction mode of current sample via intra or inter prediction can be used.

[00153] i. In one example, the intra prediction mode matrix, which is the same resolution as the reconstruction, is constructed as one input for the SR process. For each sample in the intra prediction mode matrix, the value comes from the intra prediction mode of the corresponding CU.

[00154] **Example 3**

[00155] This example involves general solutions.

[00156] 6. In one example, the above method may be applied to a specific picture/slice type, such as I slice/pictures, e.g., an NN-based SR model is trained to up-sample the reconstructed samples in I slice.

[00157] 7. In one example, the above method may be applied to B/P slice/pictures, e.g., a NN-based SR model is trained to up-sample the reconstructed samples in B slice or P slice.

[00158] **Example 4**

[00159] 1. It is proposed that for two sub-regions within a video unit (e.g., a picture/slice/tile/subpicture), two different SR methods may be applied.

[00160] a. In one example, the SR methods may include the NN-based solution.

[00161] b. In one example, the SR methods may include the non-NN-based solution (e.g., via the traditional filters).

[00162] c. In one example, for a first sub-region, the NN-based solution is used, and for a second sub-region, the non-NN-based solution is used.

[00163] d. In one example, for a first sub-region, the NN-based solution with a first design/model is used, and for a second sub-region, the NN-based solution with a second design/model is used.

[00164] i. In one example, the first/second design may have different inputs.

[00165] ii. In one example, the first/second design may have different number of layers.

[00166] iii. In one example, the first/second design may have different strides.

[00167] e. In one example, indications of the allowed SR methods and/or which SR method to be used for a sub-region may be signaled in the bitstream or derived on-the-fly.

[00168] i. In one example, it may be derived according to decoded information (e.g., how many/ratio of samples are intra coded).

[00169] ii. In one example, it may be derived according to the SR solution used for a reference sub-region (e.g., co-located sub-region).

[00170] 2. A candidate set for a video unit may be pre-defined or signaled in the bitstream wherein the candidate set may include multiple SR solutions for samples in the video unit to be chosen from.

[00171] a. In one example, the candidate set may include multiple NN-based methods with different models/designs.

[00172] b. In one example, the candidate set may include NN-based methods and non-NN-based methods.

[00173] c. In one example, different candidate sets of NN-based SR models are used for different cases, e.g., according to decoded information.

[00174] i. In one example, there are different sets of NN-based SR models corresponding to different color components, and/or different slice types, and/or different quantization parameters (QPs).

[00175] 1. In one example, QP may be categorized into several groups. For example, different NN-based SR models may be used for different group [QP/M], wherein M is an integer such as 6.

[00176] 2. In one example, the QP is fed into the SR model where one model can correspond to all the QPs. In this condition, only one QP group is used.

[00177] ii. In one example, luma component and chroma component may adopt different sets of NN-based SR models.

[00178] 1. In one example, a first set of NN-based SR models is applied to luma component, and a second set of NN-based SR models is applied to at least one chroma components.

[00179] 2. In one example, each color components is associated with its own set of NN-based SR models.

[00180] 3. Alternatively, furthermore, how many sets of NN-based SR models to be applied for the three-color components may depend on the slice/picture types, and/or partitioning tree types (single or dual tree), etc.

[00181] iii. In one example, two slice types (e.g., I slice and B (or P) slice) may utilize different sets of NN-based SR models.

[00182] iv. In one example, for a first color component, two slice types (e.g., I slice and B (or P) slice) may utilize different sets of NN-based SR models; while for a second color component, two slice types (e.g., I slice and B (or P) slice) may utilize same set of NN-based SR models.

[00183] v. In one example, for each QP or QP group, one NN-based SR model is trained. The number of NN models is equal to the number of QPs or QP groups.

[00184] 3. In one example, the NN-based (e.g. CNN-based) SR and the traditional filters can be used together.

[00185] a. In one example, for different video unit (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs) levels, different up-sampling can be used together.

[00186] i. For example, for different CTUs in one picture, some CTUs may choose the traditional filters and other CTUs may prefer the NN-based SR methods.

[00187] b. In one example, the selection of NN-based SR and the traditional filters may be signaled from the encoder to the decoder.

[00188] i. The selection may be signaled in sequence header/sequence parameter set (SPS)/picture parameter set (PPS)/Picture header/Slice header/CTU/CTB, or any rectangular region.

[00189] ii. Different selection may be signaled for different color component.

[00190] 4. In above examples, the traditional filters can be used as the up-sampling method.

[00191] a. In one example, the DCT interpolation filter (DCTIF) can be used as the up-sampling method.

[00192] b. In one example, the bilinear interpolation can be used as the up-sampling method.

[00193] c. In one example, the bi-cubic interpolation can be used as the up-sampling method.

[00194] d. In one example, the Lanczos interpolation can be used as the up-sampling method.

[00195] e. In one example, the up-sampling method may be signaled from the encoder to the decoder.

[00196] i. In one example, an index may be signaled to indicate the up-sampling filter.

[00197] ii. In one example, at least one coefficient of the up-sampling filter may be signaled, directly or indirectly.

[00198] iii. The up-sampling method may be signaled in sequence header/SPS/PPS/Picture header/Slice header/CTU/CTB, or any rectangular region.

[00199] iv. Different up-sampling method may be signaled for different color component.

[00200] f. In one example, the up-sampling method may be required by the decoder side and informed to the encoder side in an inter-active application.

[00201] 5. In one example, an NN-based SR can be used as the up-sampling method.

[00202] a. In one example, the network of the SR should include as least one up-sampling layer.

[00203] i. In one example, the neural network may be CNN.

[00204] ii. In one example, the de-convolution with a stride of K (e.g. K=2) may be used as the up-sampling layer, which is illustrated in FIG. 2.

[00205] iii. In one example, the pixel shuffling method may be used as the up-sampling layer, which is illustrated in FIG. 3.

[00206] 6. The NN-based (e.g. CNN-based) SR may be applied to certain slice/picture types, certain temporal layers, or certain slices/picture according to reference picture list information.

[00207] **Example 5**

[00208] This example involves the choice of up-sampling methods.

[00209] 7. Whether and/or how to use NN-based (e.g. CNN-based) SR (denoted as CNN information) may depend on video standard profiles or levels.

[00210] 8. Whether and/or how to use NN-based (e.g. CNN-based) SR (denoted as CNN information) may depend on color components.

[00211] 9. Whether and/or how to use NN-based (e.g. CNN-based) SR (denoted as CNN information) may depend on picture/slice type.

[00212] 10. Whether and/or how to use NN-based (e.g. CNN-based) SR (denoted as CNN information) may depend on the contents or coded information of a video unit.

[00213] a. In one example, when the variances of the reconstruction samples are greater than a predefined threshold, NN-based SR will be used.

[00214] b. In one example, when the energy of the high frequency components of the reconstruction samples is greater than a predefined threshold, NN-based SR will be used.

[00215] 11. Whether and/or how to use NN-based (e.g. CNN-based) SR (denoted as CNN information) may be controlled at a video unit (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs) level.

[00216] a. CNN information may comprise one or more of the following.

[00217] i. Indication of enabling/disabling the CNN filters.

[00218] ii. Which kind of CNN filter is applied.

[00219] iii. CNN filtering parameters.

[00220] iv. CNN models.

[00221] v. Stride for a convolutional layer.

[00222] vi. Precision of CNN parameters.

[00223] b. In one example, CNN information may be signaled in the video unit level.

[00224] i. In one example, the CNN information may be signaled in sequence header/SPS/PPS/Picture header/Slice header/CTU/CTB, or any rectangular region.

[00225] 12. The number of different CNN SR models and/or sets of CNN set models may be signaled to the decoder.

[00226] a. The number of different CNN SR models and/or sets of CNN set models may be different for different color component.

[00227] 13. In one example, a rate distortion optimization (RDO) strategy or a distortion-minimizing strategy is used to determine the up-sampling for one video unit.

[00228] a. In one example, the different CNN-based SR models will be used to up-sample the current input (for example, luma reconstruction). Then the PSNR values between the up-sampled reconstructions by different CNN-based SR models and the corresponding original input (the one which is not down-sampled and compressed) are calculated. The model which achieves the highest PSNR value will be chosen as the model for up-sampling. The index of that model may be signaled.

[00229] i. In one example, the Multi-Scale Structural Similarity (MS-SSIM) value instead of Peak Signal to Noise Ratio (PSNR) value is used as the metric.

[00230] b. In one example, the different tradition up-sampling filters are compared and the one achieves best quality metric is chosen.

[00231] i. In one example, the quality metric is PSNR.

[00232] c. In one example, the different CNN-based SR models and traditional filters are compared and the one achieves best quality metric is chosen.

[00233] i. In one example, the quality metric is PSNR.

[00234] d. The determination may be performed at encoder or at decoder.

[00235] i. If the determination is at decoder, the distortion may be calculated based on samples other than the current picture/slice//CTU/CTB, or any rectangular region.

[00236] 14. Different quality metric can be used as metric.

[00237] a. In one example, the quality metric is PSNR.

[00238] b. In one example, the quality metric is SSIM.

[00239] c. In one example, the quality metric is MS-SSIM.

[00240] d. In one example, the quality metric is Video Multi-method Assessment Fusion (VMAF).

[00241] **Example 6**

[00242] This example involves down-sampling methods of SR-based video coding.

[00243] 1. In one example, the down-sampling method can be traditional filters.

[00244] a. In one example, the Discrete Cosine Transform Interpolation Filter (DCTIF) can be used for down-sampling.

[00245] b. In one example, the bilinear interpolation can be used for down-sampling.

[00246] c. In one example, the bicubic interpolation can be used for down-sampling.

[00247] d. In one example, the down-sampling method may be signaled from the encoder to the decoder.

[00248] i. In one example, an index may be signaled to indicate the down-sampling filter.

[00249] ii. In one example, at least one coefficient of the down-sampling filter may be signaled, directly or indirectly.

[00250] iii. The down-sampling method may be signaled in sequence header/SPS/PPS/Picture header/Slice header/CTU/CTB, or any rectangular region.

[00251] iv. Different down-sampling method may be signaled for different color component.

[00252] e. In one example, the down-sampling method may be required by the decoder side and informed to the encoder side in an inter-active application.

[00253] 2. In one example, the down-sampling method can be neural network (NN) based such as convolutional neural network (CNN) based, method.

[00254] a. The CNN-based down-sampling method should include at least one down-sampling layer.

[00255] i. In one example, the convolution with stride of K (e.g. K=2) can be used as the down-sampling layer and the down-sampling ratio is K.

[00256] ii. In one example, the pixel-unshuffling method followed by a convolution with stride of 1 can be used for down-sampling. The pixel-unshuffling is illustrated in FIG. 7. FIG. 7 is a schematic diagram of an example of an inverse process of the pixel shuffle 700. In an embodiment, the inverse process of the pixel shuffle has a down-sampling ratio of 2.

[00257] 3. A series of down-sampling can be used for achieving a specific down-sampling ratio.

[00258] a. In one example, two convolution layers with stride of K (e.g., K=2) is used in one network. In this condition, the down-sampling ratio is 4.

[00259] b. In one example, two traditional down-sampling filters (down-sampling ratio of each is 2) are used for a down-sampling ratio of 4.

[00260] 4. In one example, the traditional filters and the CNN-based methods can be combined for a specific down-sampling ratio.

[00261] a. In one example, the traditional filters is used followed by a CNN-based method. The traditional filter achieves 2x down-sampling and the CNN-based method achieves 2x down-sampling. Thus, the input is down-sampled by 4x.

[00262] 5. When down-sample a specific input video unit level, Different down-sampling methods may be compared with each other to choose a best down-sampling method.

[00263] a. In one example, there are K (e.g. K=3) CNN-based down-sampling models. For one specific input, the three down-sampling models will down-sample the input, respectively. The down-sampled reconstruction will be up-sampled to the original resolution. The quality metric (e.g., PSNR) is utilized to measure the three up-sampled results. The model who achieves the best performance will be utilized to perform the real down-sampling.

[00264] i. In one example, the quality metric is MS-SSIM.

[00265] ii. In one example, the quality metric is PSNR.

[00266] b. The index of the down-sampling methods may be signaled to the encoder or decoder.

[00267] 6. The down-sampling methods may be signaled to the decoder.

[00268] a. In one example, the CNN-based down-sampling methods are used for down-sampling. For one specific video unit (e.g. frame) level, the index of chosen model will be signaled to the decoder.

[00269] b. In one example, different CTUs within one frame use different down-sampling methods. In this condition, all the index of the corresponding methods may be signaled to the decoder.

[00270] c. In one example, at least one coefficient of the down-sampling filter may be signaled, directly or indirectly.

[00271] d. Different down-sampling method may be signaled for different color component.

[00272] e. In one example, the down-sampling method may be required by the decoder side and informed to the encoder side in an inter-active application.

[00273] **Example 7**

[00274] This example discusses the down-sampling ratio of input.

[00275] 7. The input of down-sampling methods can be at all the video unit (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs) level.

[00276] a. In one example, the input is the frame level with size of its original resolution.

[00277] b. In one example, the input is one CTU level with size of 128x128.

[00278] 8. In one example, the input is a block within one frame whose size is not limited.

[00279] c. In one example, the input can be a block with spatial size (M, N), for example, M=256, N=128.

[00280] 9. In one example, the down-sampling ratio can be different for all the video unit (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs) levels.

[00281] d. In one example, the down-sample ratio is 2 for all the frames of one sequence.

[00282] e. In one example, the down-sample ratio is 2 for all the CTUs of one frame.

[00283] f. In one example, the down-sample ratio is 2 for the first frame and it may be 4 for the next frame.

[00284] g. The combination of down-sampling ratio for different video unit levels may be used.

[00285] i. In one example, the down-sample ratio is 2 for one frame and it may be 4 for one CTU in the same frame. In the condition, the CTU will be down-sampled by 4x.

[00286] 10. In one example, the down-sampling ratio can be different for all the components of the input video unit level.

[00287] h. In one example, the down-sampling ratio is 2 for both luma and chroma components.

[00288] i. In another example, the down-sampling ratio is 2 for luma component and it is 4 for chroma components.

[00289] 11. In one example, the down-sampling ratio can be 1 which means no down-sampling is performed.

[00290] j. The down-sampling ratio can be applied at all the video unit (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs) level.

[00291] 12. The down-sampling ratio can be determined by comparison.

[00292] k. In one example, there are 2x and 4x down-sampling ratio for one frame can be used. In this condition, the encoder may compress it with 2x down-sampling and then compressed it with 4x down-sampling. After that, then up-sample the low-resolution reconstruction with the

same up-sampling method. Then calculate the quality metric (e.g., PSNR) of each results. The down-sampling ratio which achieves the best reconstruction quality will be chosen as the real down-sampling ratio for compression.

[00293] i. In one example, the quality metric is MS-SSIM.

[00294] 13. The determination may be performed at encoder or at decoder.

[00295] ii. If the determination is at decoder, the distortion may be calculated based on samples other than the current picture/slice//CTU/CTB, or any rectangular region.

[00296] 14. Different quality metric can be used as metric for the comparison.

[00297] e. In one example, the quality metric is PSNR.

[00298] f. In one example, the quality metric is SSIM.

[00299] g. In one example, the quality metric is MS-SSIM.

[00300] h. In one example, the quality metric is VMAF.

[00301] 15. In one example, the down-sampling ratio may be signaled in the video unit level.

[00302] l. In one example, the CNN information may be signaled in SPS/PPS/Picture header/Slice header/CTU/CTB.

[00303] **Example 8**

[00304] This example involves the position of SR. FIGS. 6A-6D are schematic diagrams illustrating examples of different positions for up-sampling 600.

[00305] 1. The super resolution (SR) process such as NN-based or Non-NN-based SR process may be placed before in-loop filters.

[00306] a. In one example, the SR process may be invoked right after a block (e.g., a CTU/CTB) is reconstructed.

[00307] b. In one example, the SR process may be invoked right after a region (e.g., a CTU row) is reconstructed.

[00308] 2. The super resolution (SR) process such as NN-based or Non-NN-based SR process may be placed in different locations in the chain of in-loop filters.

[00309] a. In one example, the SR process may be applied before or after a given in-loop filters.

[00310] i. In one example, the SR process is placed before the deblocking filter (DBF) as illustrated in FIG. 6A.

[00311] ii. In one example, the SR process is placed between DBF and SAO as illustrated in FIG. 6B.

[00312] iii. In one example, the SR process is placed between SAO and ALF as illustrated in FIG. 6C.

[00313] iv. In one example, the super resolution is placed after ALF as illustrated in FIG. 6D.

[00314] v. In one example, the SR process is placed before SAO.

[00315] vi. In one example, the SR process is placed before ALF.

[00316] b. In one example, whether to apply SR before a given in-loop filter may depend on whether the loop-filter decision process is taking the original image into consideration.

[00317] 3. Indication of the position of SR process may be signaled in the bitstream or determined on-the-fly according to decoded information.

[00318] 4. The SR process such as NN-based or Non-NN-based SR process may be exclusively used with other coding tools such as in-loop filters, i.e., when the SR process is applied, then one or multiple kinds of the in-loop filters may not be applied any more, or vice versa.

[00319] a. In one example, the SR process may be used exclusively with at least one kind of in-loop filters.

[00320] i. In one example, the original loop filters, such as DB, SAO, and ALF are all turned off when the SR process is applied.

[00321] ii. In one example, the SR process may be applied when ALF is disabled.

[00322] iii. In one example, the SR process may be applied to chroma components when Cross Component Adaptive Loop Filter (CC-ALF) is disabled.

[00323] b. In one example, signaling of side information of an in-loop filtering method may be dependent on whether/how the SR process is applied.

[00324] c. In one example, whether/how the SR process is applied may be dependent on the usage of an in-loop filtering method.

[00325] **Example 9**

[00326] This example involves the SR network structure.

[00327] 5. The proposed NN-based (e.g., CNN-based) SR network comprises multiple convolutional layers. There is an up-sampling layer used in the proposed network to up-sample the resolution.

[00328] a. In one example, the de-convolution with stride of K greater than 1 (e.g. K = 2) can be used for up-sampling.

[00329] vi. In one example, K may be dependent on decoded information (e.g., color format).

[00330] b. In one example, the pixel shuffling is used for up-sampling as shown in FIG. 4. FIG. 4 is a schematic diagram of an example for an up-sampling network 400. Suppose the down-sampling ratio is K where the resolution of LR input is 1/K of the original input. The first 3x3 convolution is used to fuse the information from LR input and generate the feature maps.

The output feature maps from the first convolutional layer then go through several sequentially stacked residual blocks, each labeled RB. Feature maps are labeled M and R. The last convolutional layer takes the feature maps from the last residual block as input and produces R (e.g., R=K*K) feature maps. Finally, a shuffle layer is adopted to generate the filter image whose spatial resolution is the same with the original resolution.

[00331] c. In one example, the residual blocks may be used in the SR network. In one example, the residual blocks consists of three sequentially connected components as shown in FIG. 5: one convolutional layer, one PReLU activation function, and a convolutional layer. The input to the first convolutional layer is added to the output of the second convolutional layer.

[00332] 6. The inputs of the NN-based (e.g. CNN-based) SR network can be different video units (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs, or any rectangular region) levels.

[00333] a. In one example, the input of SR network can be a CTU block which is down-sampled.

[00334] b. In one example, the input is the whole frame which is down-sampled.

[00335] 7. The input of NN-based (e.g. CNN-based) SR network may be a combination of different color components.

[00336] a. In one example, the input may be the luma component of reconstruction.

[00337] b. In one example, the input may be the chroma components of reconstruction.

[00338] c. In one example, the input may be both luma and chroma components of the same reconstruction.

[00339] 8. In one example, the luma component may be used as the input and the output of the NN-based (e.g. CNN-based) SR network is the up-sampled chroma components.

[00340] 9. In one example, the chroma components may be used as the input and the output of the NN-based (e.g. CNN-based) SR network is the up-sampled luma component.

[00341] 10. The NN-based (e.g. CNN-based) SR network is not limited to up-sample the reconstructions.

[00342] a. In one example, the decoded side information may be used as the input of NN-based (e.g. CNN-based) SR network for up-sampling.

[00343] i. In one example, the prediction picture may be used as the input for up-sampling. The output of the network is the up-sampled prediction picture.

[00344] **Example 10**

[00345] This example involves the processing unit of SR.

[00346] 1. Super resolution/up-sampling process may be performed at a SR unit level wherein the SR unit covers more than one sample/pixel.

[00347] a. In one example, the SR unit may be the same as the video unit wherein down-sampling process is invoked.

[00348] b. In one example, the SR unit may be different from the video unit wherein down-sampling process is invoked.

[00349] i. In one example, even the down-sampling is performed in the picture/slice/tile level, the SR unit may be a block (e.g., a CTU).

[00350] ii. In one example, even the down-sampling is performed in the CTU/CTB level, the SR unit may be CTU row or multiple CTU/CTBs.

[00351] c. Alternatively, furthermore, for the NN-based SR methods, the inputs to the network may be set to the SR unit.

[00352] d. Alternatively, furthermore, for the NN-based SR methods, the inputs to the network may be set to a region containing the SR unit to be up-sampled and other samples/pixels.

[00353] e. In one example, the SR unit may be indicated in a bitstream or pre-defined.

[00354] 2. For two SR units, the super resolution methods/up-sampling methods may be different.

[00355] a. In one example, the super resolution methods/up-sampling methods may include the NN-based solution and the non-NN-based solution (e.g., traditional up-sampling filtering methods).

[00356] 3. The inputs of SR network can be at different video units (e.g., sequence/picture/slice/tile/brick/subpicture/CTU/CTU row/one or multiple CUs or CTUs/CTBs, or any region covers more than one sample/pixel) level.

[00357] a. In one example, the input of SR network can be a CTU block which is down-sampled.

[00358] b. In one example, the input is the whole frame which is down-sampled.

[00359] 4. The CNN-based SR models can be used to up-sample the different video unit level.

[00360] a. In one example, the CNN-based SR models are trained on the frame-level data and is used to up-sample the frame-level input.

[00361] b. In one example, the CNN-based SR models are trained on the frame-level data and is used to up-sample the CTU-level input.

[00362] c. In one example, the CNN-based SR models are trained on the CTU-level data and is used to up-sample the frame-level input.

[00363] d. In one example, the CNN-based SR models are trained on the CTU-level data and is used to up-sample the CTU-level input.

[00364] **Example 11**

[00365] This example involves the side information for input of SR network.

[00366] 5. The down-sampling ratio of a video unit may be treated as inputs of the SR network.

[00367] a. Alternatively, furthermore, the convolution layer may be configured with a stride which is dependent on the down sampling ratio.

[00368] b. The down-sampling ratio for the input of SR network can be any positive integers.

[00369] i. Alternatively, furthermore, and the minimal spatial resolution of the input shall be 1x1.

[00370] c. The down-sampling ratio for the input of SR network may be a ratio of any two positive integers, such as 3:2.

[00371] d. The horizontal down-sampling ratio and vertical down-sampling ratio may be the same, or they may be different.

[00372] 6. It is proposed that the encoded/decoded information can be utilized during the up-sampling process.

[00373] a. In one example, the encoded/decoded information may be used as the inputs of the super resolution network.

[00374] b. In one example, the encoded/decoded information may include but not limited to prediction signal, partition structure, intra prediction mode.

[00375] Other technical solutions are discussed.

[00376] FIG. 8 is a schematic diagram of an overall framework for up-sampling 800 according to an embodiment of the disclosure. The preprocess will extract and normalize the QP value, Y, U, and V data from the reconstruction rec_{LR} and the prediction $pred_{LR}$. The output of pre-processing is the normalized QP map and Y, U, and V data used for the up-sampling. A neural network based super resolution technology is utilized for up-sampling. In the present disclosure, two neural networks are used. One network is designed for the Y channel data and the other is designed for U and V channel data. The Y_{HR}^{rec}, U_{HR}^{rec}, and V_{HR}^{rec} are the up-sampled reconstruction data. The final up-sampled reconstruction is consisted of de-normalized Y_{HR}^{rec}, U_{HR}^{rec}, and V_{HR}^{rec}.

[00377] The pre-processing before upsampling is discussed.

[00378] 1. Normalize the reconstruction Y, U, and V channels, the prediction Y channel, and the base QP value. In one example, the equation for normalization is:

$$Y_{LR}^{rec}[i,j] = \frac{Y^{rec}[i,j]}{1023.0}$$
$$U_{LR}^{rec}[i,j] = \frac{U^{rec}[i,j]}{1023.0}$$
$$V_{LR}^{rec}[i,j] = \frac{V^{rec}[i,j]}{1023.0}$$

$$Y_{LR}^{pred}[i,j] = \frac{Y^{pred}[i,j]}{1023.0}$$
$$QP_{norm} = \frac{QP}{51.0}$$

where the [i, j] is the coordinate of a pixel in the frame, the Y^{rec} denotes the Y channel of the reconstruction frame, the U^{rec} denotes the U channel of the reconstruction frame, the V^{rec} denotes the V channel of the reconstruction frame, and Y^{pred} denotes the Y channel of the prediction frame.

[00379] Construct the QP map by filling a matrix with QP_{norm} and the size of QP map should be the same with Y_{LR}^{rec}.

$$QP_MAP[i,j] = QP_{norm}$$

where [i, j] is the coordinate of a pixel in the frame.

[00380] The up-sampling for Y channel is discussed.

[00381] Feed the QP_MAP, Y_{LR}^{pred}, and Y_{LR}^{rec} into the neural network designed for Y channel. In one example, the neural network is illustrated in FIG. 9. FIG. 9 is a schematic diagram of an example of a NN for reconstruction of a Y channel 900. The term "Conv kxk, M" denotes the convolution with kernel size, k, and the number of filters is M. The term B is the number of residual blocks (RBs). The term R is the square of scaling ratio, r.

[00382] The input of network consists of three parts: QP map, Y_{LR}^{rec}, and Y_{LR}^{pred}. The QP map is the base QP for compression and the Y_{LR}^{rec} and Y_{LR}^{pred} denote the low-resolution reconstruction frame and the corresponding low-resolution prediction frame, respectively. The Y_{HR}^{rec} denotes the high-resolution output of neural network, which is the same resolution with the original frame. As shown in FIG. 9, the network is composed of residual blocks and a pixel shuffle layer for up-sampling.

[00383] In one example, residual blocks are shown in FIG. 5. In one example, the up-sampling block in FIG. 9 uses the pixel shuffle layer as illustrated in FIG. 10. FIG. 10 is a schematic diagram of an example of a pixel shuffle operator 1000. A low-resolution block (LR block) with size 2x2x4 is upsampled to a high-resolution block (HR block) with size 4x4x1. The

font of the first element in each channel of the LR block and the corresponding position in the HR block are bold.

[00384] Alternatively, the up-sampling block can use the de-convolution with stride equal to the up-scaling ratio.

[00385] In another example, the body of neural network can be different, as long as it has one up-sampling layer before the output of neural network.

[00386] De-normalize the output of the neural network designed for Y channel data. In one example, the equation for de-normalizing is:

$$Y_{HR-denorm}^{rec}[i,j] = \lfloor 1023.0 * Y_{HR}^{rec}[i,j] + 0.5 \rfloor$$

where [i, j] is the coordinate of a pixel in the frame, Y_{HR}^{rec} is the output of the neural network, and $\lfloor x \rfloor$ returns the floor of input x.

[00387] The following steps are used for up-sampling chroma components (U and V channels).

[00388] 5. Down-sample the Y_{LR}^{rec} to the same resolution with U_{LR}^{rec}. The down-sampled Y_{LR}^{rec} is denoted as $Y_{LR-down}^{rec}$.

[00389] a) In one example, the down-sample method is bi-linear interpolation.

[00390] b) In one example, the down-sample method is bi-cubic interpolation.

[00391] c) In one example, the down-sample method is convolution with stride equal to the scaling ratio for original frame.

[00392] d) In one example, the down-sample method is the inverse of pixel shuffle 700, which is illustrated in FIG. 7.

[00393] 6. Feed $Y_{LR-down}^{rec}$, U_{LR}^{rec}, V_{LR}^{rec}, and QP_MAP into the neural network designed for U and V channels. In one example, the neural network is illustrated in FIG. 11. FIG. 11 is a schematic diagram of an example of a neural network for reconstruction of U and V channels 1100. In FIG. 11, the term Down denotes the down-sampling. The term "Conv kxk, M" denotes the convolution with kernel size, k, and the number of filters is M. The term B is the number of residual blocks (RBs). The term R is the square of scaling ratio, r. The input of network consists of four parts: QP map, Y_{LR}^{rec}, U_{LR}^{rec}, and U_{LR}^{rec}. The QP map is the base QP for compression, and the Y_{LR}^{rec}, U_{LR}^{rec}, and U_{LR}^{rec} denote the low-resolution reconstruction frame for Y, U, and V channels, respectively. The U_{HR}^{rec} and V_{HR}^{rec} denotes the high-resolution reconstruction of U and V channels, respectively.

[00394] a) In one example, residual blocks are shown in FIG. 5.

[00395] b) In one example, the up-sampling layer is the pixel shuffle layer.

[00396] c) Alternatively, the up-sampling block can use the de-convolution with stride equal to the up-scaling ratio.

[00397] d) In another example, the body of neural network can be different, as long as it has one up-sampling layer before the output of neural network.

[00398] FIG. 12 is a block diagram showing an example video processing system 1200 in which various techniques disclosed herein may be implemented. Various implementations may include some or all of the components of the video processing system 1200. The video processing system 1200 may include input 1202 for receiving video content. The video content may be received in a raw or uncompressed format, e.g., 8 or 10 bit multi-component pixel values, or may be in a compressed or encoded format. The input 1202 may represent a network interface, a peripheral bus interface, or a storage interface. Examples of network interface include wired interfaces such as Ethernet, passive optical network (PON), etc. and wireless interfaces such as Wi-Fi or cellular interfaces.

[00399] The video processing system 1200 may include a coding component 1204 that may implement the various coding or encoding methods described in the present document. The coding component 1204 may reduce the average bitrate of video from the input 1202 to the output of the coding component 1204 to produce a coded representation of the video. The coding techniques are therefore sometimes called video compression or video transcoding techniques. The output of the coding component 1204 may be either stored, or transmitted via a communication connected, as represented by the component 1206. The stored or communicated bitstream (or coded) representation of the video received at the input 1202 may be used by the component 1208 for generating pixel values or displayable video that is sent to a display interface 1210. The process of generating user-viewable video from the bitstream representation is sometimes called video decompression. Furthermore, while certain video processing operations are referred to as "coding" operations or tools, it will be appreciated that the coding tools or operations are used at an encoder and corresponding decoding tools or operations that reverse the results of the coding will be performed by a decoder.

[00400] Examples of a peripheral bus interface or a display interface may include universal serial bus (USB) or high definition multimedia interface (HDMI) or Displayport, and so on. Examples of storage interfaces include SATA (serial advanced technology attachment), Peripheral Component Interconnect (PCI), Integrated Drive Electronics (IDE) interface, and the like. The techniques described in the present document may be embodied in various electronic devices such as mobile phones, laptops, smartphones or other devices that are capable of performing digital data processing and/or video display.

[00401] FIG. 13 is a block diagram of a video processing apparatus 1300. The video processing apparatus 1300 may be used to implement one or more of the methods described herein. The video processing apparatus 1300 may be embodied in a smartphone, tablet, computer, Internet of Things (IoT) receiver, and so on. The video processing apparatus 1300 may include one or more processors 1302, one or more memories 1304 and video processing hardware 1306 (a.k.a., video processing circuitry). The processor(s) 1302 may be configured to implement one or more methods described in the present document. The memory (memories) 1304 may be used for storing data and code used for implementing the methods and techniques described herein. The video processing hardware 1306 may be used to implement, in hardware circuitry, some techniques described in the present document. In some embodiments, the video processing hardware 1306 may be partly or completely located within the processor 1302, e.g., a graphics processor.

[00402] FIG. 14 is a block diagram that illustrates an example of a video coding system 1400 that may utilize the techniques of this disclosure. As shown in FIG. 14, the video coding system 1400 may include a source device 1410 and a destination device 1420. Source device 1410 generates encoded video data which may be referred to as a video encoding device. Destination device 1420 may decode the encoded video data generated by source device 1410 which may be referred to as a video decoding device.

[00403] Source device 1410 may include a video source 1412, a video encoder 1414, and an input/output (I/O) interface 1416.

[00404] Video source 1412 may include a source such as a video capture device, an interface to receive video data from a video content provider, and/or a computer graphics system for generating video data, or a combination of such sources. The video data may comprise one or more pictures. Video encoder 1414 encodes the video data from video source 1412 to generate a bitstream. The bitstream may include a sequence of bits that form a coded representation of the video data. The bitstream may include coded pictures and associated data. The coded picture is a coded representation of a picture. The associated data may include sequence parameter sets, picture parameter sets, and other syntax structures. I/O interface 1416 may include a modulator/demodulator (modem) and/or a transmitter. The encoded video data may be transmitted directly to destination device 1420 via I/O interface 1416 through network 1430. The encoded video data may also be stored onto a storage medium/server 1440 for access by destination device 1420.

[00405] Destination device 1420 may include an I/O interface 1426, a video decoder 1424, and a display device 1422.

[00406] I/O interface 1426 may include a receiver and/or a modem. I/O interface 1426 may acquire encoded video data from the source device 1410 or the storage medium/server 1440. Video decoder 1424 may decode the encoded video data. Display device 1422 may display the decoded video data to a user. Display device 1422 may be integrated with the destination device 1420, or may be external to destination device 1420 which may be configured to interface with an external display device.

[00407] Video encoder 1414 and video decoder 1424 may operate according to a video compression standard, such as the High Efficiency Video Coding (HEVC) standard, Versatile Video Coding (VVC) standard, and other current and/or further standards.

[00408] FIG. 15 is a block diagram illustrating an example of a video encoder 1500, which may be video encoder 1414 in the video coding system 1400 illustrated in FIG. 14.

[00409] Video encoder 1500 may be configured to perform any or all of the techniques of this disclosure. In the example of FIG. 15, video encoder 1500 includes a plurality of functional components. The techniques described in this disclosure may be shared among the various components of video encoder 1500. In some examples, a processor may be configured to perform any or all of the techniques described in this disclosure.

[00410] The functional components of video encoder 1500 may include a partition unit 1501, a prediction unit 1502 which may include a mode selection unit 1503, a motion estimation unit 1504, a motion compensation unit 1505 and an intra prediction unit 1506, a residual generation unit 1507, a transform unit 1508, a quantization unit 1509, an inverse quantization unit 1510, an inverse transform unit 1511, a reconstruction unit 1512, a buffer 1513, and an entropy encoding unit 1514.

[00411] In other examples, video encoder 1500 may include more, fewer, or different functional components. In an example, prediction unit 1502 may include an intra block copy (IBC) unit. The IBC unit may perform prediction in an IBC mode in which at least one reference picture is a picture where the current video block is located.

[00412] Furthermore, some components, such as motion estimation unit 1504 and motion compensation unit 1505 may be highly integrated, but are represented in the example of FIG. 15 separately for purposes of explanation.

[00413] Partition unit 1501 may partition a picture into one or more video blocks. Video encoder 1414 and video decoder 1424 of FIG. 14 may support various video block sizes.

[00414] Mode selection unit 1503 may select one of the coding modes, intra or inter, e.g., based on error results, and provide the resulting intra- or inter-coded block to a residual generation unit 1507 to generate residual block data and to a reconstruction unit 1512 to

reconstruct the encoded block for use as a reference picture. In some examples, mode selection unit 1503 may select a combination of intra and inter prediction (CIIP) mode in which the prediction is based on an inter prediction signal and an intra prediction signal. Mode selection unit 1503 may also select a resolution for a motion vector (e.g., a sub-pixel or integer pixel precision) for the block in the case of inter-prediction.

[00415] To perform inter prediction on a current video block, motion estimation unit 1504 may generate motion information for the current video block by comparing one or more reference frames from buffer 1513 to the current video block. Motion compensation unit 1505 may determine a predicted video block for the current video block based on the motion information and decoded samples of pictures from buffer 1513 other than the picture associated with the current video block.

[00416] Motion estimation unit 1504 and motion compensation unit 1505 may perform different operations for a current video block, for example, depending on whether the current video block is an I slice, a P slice, or a B slice. I-slices (or I-frames) are the least compressible but don't require other video frames to decode. P-slices (or P-frames) can use data from previous frames to decompress and are more compressible than I-frames. B-slices (or B-frames) can use both previous and forward frames for data reference to get the highest amount of data compression.

[00417] In some examples, motion estimation unit 1504 may perform uni-directional prediction for the current video block, and motion estimation unit 1504 may search reference pictures of list 0 or list 1 for a reference video block for the current video block. Motion estimation unit 1504 may then generate a reference index that indicates the reference picture in list 0 or list 1 that contains the reference video block and a motion vector that indicates a spatial displacement between the current video block and the reference video block. Motion estimation unit 1504 may output the reference index, a prediction direction indicator, and the motion vector as the motion information of the current video block. Motion compensation unit 1505 may generate the predicted video block of the current block based on the reference video block indicated by the motion information of the current video block.

[00418] In other examples, motion estimation unit 1504 may perform bi-directional prediction for the current video block, motion estimation unit 1504 may search the reference pictures in list 0 for a reference video block for the current video block and may also search the reference pictures in list 1 for another reference video block for the current video block. Motion estimation unit 1504 may then generate reference indexes that indicate the reference pictures in list 0 and list 1 containing the reference video blocks and motion vectors that indicate spatial displacements

between the reference video blocks and the current video block. Motion estimation unit 1504 may output the reference indexes and the motion vectors of the current video block as the motion information of the current video block. Motion compensation unit 1505 may generate the predicted video block of the current video block based on the reference video blocks indicated by the motion information of the current video block.

[00419] In some examples, motion estimation unit 1504 may output a full set of motion information for decoding processing of a decoder.

[00420] In some examples, motion estimation unit 1504 may not output a full set of motion information for the current video. Rather, motion estimation unit 1504 may signal the motion information of the current video block with reference to the motion information of another video block. For example, motion estimation unit 1504 may determine that the motion information of the current video block is sufficiently similar to the motion information of a neighboring video block.

[00421] In one example, motion estimation unit 1504 may indicate, in a syntax structure associated with the current video block, a value that indicates to the video decoder 1424 of FIG. 14 that the current video block has the same motion information as another video block.

[00422] In another example, motion estimation unit 1504 may identify, in a syntax structure associated with the current video block, another video block and a motion vector difference (MVD). The motion vector difference indicates a difference between the motion vector of the current video block and the motion vector of the indicated video block. The video decoder 1424 of FIG. 14 may use the motion vector of the indicated video block and the motion vector difference to determine the motion vector of the current video block.

[00423] As discussed above, video encoder 1414 of FIG. 14 may predictively signal the motion vector. Two examples of predictive signaling techniques that may be implemented by video encoder 1414 of FIG. 14 include advanced motion vector prediction (AMVP) and merge mode signaling.

[00424] Intra prediction unit 1506 may perform intra prediction on the current video block. When intra prediction unit 1506 performs intra prediction on the current video block, intra prediction unit 1506 may generate prediction data for the current video block based on decoded samples of other video blocks in the same picture. The prediction data for the current video block may include a predicted video block and various syntax elements.

[00425] Residual generation unit 1507 may generate residual data for the current video block by subtracting (e.g., indicated by the minus sign) the predicted video block(s) of the current video block from the current video block. The residual data of the current video block may include

residual video blocks that correspond to different sample components of the samples in the current video block.

[00426] In other examples, there may be no residual data for the current video block, for example in a skip mode, and residual generation unit 1507 may not perform the subtracting operation.

[00427] Transform unit 1508 may generate one or more transform coefficient video blocks for the current video block by applying one or more transforms to a residual video block associated with the current video block.

[00428] After transform unit 1508 generates a transform coefficient video block associated with the current video block, quantization unit 1509 may quantize the transform coefficient video block associated with the current video block based on one or more quantization parameter (QP) values associated with the current video block.

[00429] Inverse quantization unit 1510 and inverse transform unit 1511 may apply inverse quantization and inverse transforms to the transform coefficient video block, respectively, to reconstruct a residual video block from the transform coefficient video block. Reconstruction unit 1512 may add the reconstructed residual video block to corresponding samples from one or more predicted video blocks generated by the prediction unit 1502 to produce a reconstructed video block associated with the current block for storage in the buffer 1513.

[00430] After reconstruction unit 1512 reconstructs the video block, loop filtering operation may be performed to reduce video blocking artifacts in the video block.

[00431] Entropy encoding unit 1514 may receive data from other functional components of the video encoder 1500. When entropy encoding unit 1514 receives the data, entropy encoding unit 1514 may perform one or more entropy encoding operations to generate entropy encoded data and output a bitstream that includes the entropy encoded data.

[00432] FIG. 16 is a block diagram illustrating an example of a video decoder 1600, which may be video decoder 1424 in the video coding system 1400 illustrated in FIG. 14.

[00433] The video decoder 1600 may be configured to perform any or all of the techniques of this disclosure. In the example of FIG. 16, the video decoder 1600 includes a plurality of functional components. The techniques described in this disclosure may be shared among the various components of the video decoder 1600. In some examples, a processor may be configured to perform any or all of the techniques described in this disclosure.

[00434] In the example of FIG. 16, video decoder 1600 includes an entropy decoding unit 1601, a motion compensation unit 1602, an intra prediction unit 1603, an inverse quantization unit 1604, an inverse transformation unit 1605, a reconstruction unit 1606 and a buffer 1607.

Video decoder 1600 may, in some examples, perform a decoding pass generally reciprocal to the encoding pass described with respect to video encoder 1414 (FIG. 14).

[00435] Entropy decoding unit 1601 may retrieve an encoded bitstream. The encoded bitstream may include entropy coded video data (e.g., encoded blocks of video data). Entropy decoding unit 1601 may decode the entropy coded video data, and from the entropy decoded video data, motion compensation unit 1602 may determine motion information including motion vectors, motion vector precision, reference picture list indexes, and other motion information. Motion compensation unit 1602 may, for example, determine such information by performing the AMVP and merge mode signaling.

[00436] Motion compensation unit 1602 may produce motion compensated blocks, possibly performing interpolation based on interpolation filters. Identifiers for interpolation filters to be used with sub-pixel precision may be included in the syntax elements.

[00437] Motion compensation unit 1602 may use interpolation filters as used by video encoder 1414 during encoding of the video block to calculate interpolated values for sub-integer pixels of a reference block. Motion compensation unit 1602 may determine the interpolation filters used by video encoder 1414 according to received syntax information and use the interpolation filters to produce predictive blocks.

[00438] Motion compensation unit 1602 may use some of the syntax information to determine sizes of blocks used to encode frame(s) and/or slice(s) of the encoded video sequence, partition information that describes how each macroblock of a picture of the encoded video sequence is partitioned, modes indicating how each partition is encoded, one or more reference frames (and reference frame lists) for each inter-encoded block, and other information to decode the encoded video sequence.

[00439] Intra prediction unit 1603 may use intra prediction modes for example received in the bitstream to form a prediction block from spatially adjacent blocks. Inverse quantization unit 1604 inverse quantizes, i.e., de-quantizes, the quantized video block coefficients provided in the bitstream and decoded by entropy decoding unit 1601. Inverse transform unit 1605 applies an inverse transform.

[00440] Reconstruction unit 1606 may sum the residual blocks with the corresponding prediction blocks generated by motion compensation unit 1602 or intra-prediction unit 1603 to form decoded blocks. If desired, a deblocking filter may also be applied to filter the decoded blocks in order to remove blockiness artifacts. The decoded video blocks are then stored in buffer 1607, which provides reference blocks for subsequent motion compensation/intra prediction and also produces decoded video for presentation on a display device.

[00441] FIG. 17 is a method 1700 of processing video data according to an embodiment of the disclosure. The method 1700 may be performed by a coding apparatus (e.g., an encoder) having a processor and a memory. The method 1700 may be implemented when SR or up-sampling is desired.

[00442] In block 1702, the coding apparatus uses coded information during application of a super resolution (SR) process to a video unit

[00443] In block 1704, the coding apparatus performing a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information. When implemented in an encoder, converting includes receiving a video file (e.g., a video unit) and encoding the video file into a bitstream. When implemented in a decoder, converting includes receiving a bitstream including the video file, and decoding the bitstream to obtain the video file.

[00444] In an embodiment, the coded information comprises encoded information. In an embodiment, the coded information comprises decoded information.

[00445] In an embodiment, the SR process comprises a neural network (NN)-based SR process, and wherein the coded information is used as an input to the NN-based SR process. In an embodiment, the coded information is used to determine that the SR process, which is one of a plurality of SR processes available, is applied.

[00446] In an embodiment, the coded information comprises one or more of partition information, prediction information, and an intra prediction mode.

[00447] In an embodiment, an input of the SR process comprises reconstructed low-resolution samples and decoded information, and wherein the decoded information comprises one or more of partition information, prediction information, and an intra prediction mode. In an embodiment, low resolution samples are samples prior to up-sampling. In an embodiment, high resolution samples are samples after up-sampling. In an embodiment, low-resolution samples have a resolution less than that of the high-resolution samples.

[00448] In an embodiment, the coded information comprises partition information having a same resolution as a reconstructed low-resolution frame.

[00449] In an embodiment, the method further comprises deriving sample values in the partition information by averaging reconstructed samples in the video unit.

[00450] In an embodiment, the coded information comprises prediction information, and wherein the prediction information comprises prediction samples from intra prediction, intra block copy (IBC), or inter-prediction. Intra-prediction takes advantage of redundancy in the information of neighboring pixel values within video frames to predict blocks of pixels from

their surrounding pixels and thus allowing to transmit the prediction errors instead of the pixel values themselves. IBC allows for the prediction of a given intra coded block to be a copy of another intra coded block in the same frame (i.e. from the reconstructed part of the current frame). Inter prediction creates a prediction model from one or more previously encoded video frames.

[00451] In an embodiment, the coded information comprises an intra prediction mode having a same resolution as a reconstructed low-resolution frame. In VVC, there are 67 prediction modes for a square coding unit (CU). In particular, there are 65 directional prediction modes in addition to a planar mode and a direct current (DC) mode. A CU comprises a coding block of luma samples, two corresponding coding blocks of chroma samples of a picture that has three sample arrays in the single tree mode, or a coding block of luma samples of a picture that has three sample arrays in the dual tree mode, or two coding blocks of chroma samples of a picture that has three sample arrays in the dual tree mode, or a coding block of samples of a monochrome picture, and syntax structures used to code the samples. A coding block comprises an MxN block of samples for some values of M and N such that the division of a CTB into coding blocks is a partitioning.

[00452] In an embodiment, the method further comprises deriving sample values in the intra prediction mode by filling the intra prediction mode in a corresponding video unit.

[00453] In an embodiment, the coded information comprises quantization parameter (QP) information, and wherein the QP information is used to generate an up-sampled reconstruction. Residuals are transformed into the spatial frequency domain by an integer transform that approximates the familiar Discrete Cosine Transform (DCT). The QP determines the step size for associating the transformed coefficients with a finite set of steps.

[00454] In an embodiment, the method further comprises filling a matrix with quantization (QP) parameter values to construct a QP map, wherein the QP map is used as an input to the SR process, and wherein a spatial resolution of the QP map is the same as the spatial resolution of another input of the SR process.

[00455] In an embodiment, the video unit comprises a first color component, and wherein a second color component is utilized during the SR process applied to the first color component. In an embodiment, a color component is a component within the YUV model, the red/green/blue (RGB) model, etc. The YUV model defines a color space in terms of one luminance and two chrominance components. Y stands for the luminance component (the brightness) and U and V are the chrominance (color) components, specifically the blue-difference and red-difference chroma components.

[00456] In an embodiment, the video unit comprises a first color component, and wherein a second color component is used as an input to the SR process applied to the first color component.

[00457] In an embodiment, the SR process comprises a luma up-sampling process, and wherein chroma information is used as an input to the luma up-sampling process. In an embodiment, the SR process comprises a chroma up-sampling process, and wherein luma information is used as an input to the chroma up-sampling process. In an embodiment, the luma information comprises luma samples of reconstruction, and wherein the luma samples of reconstruction are used as an input to the SR process after in-loop filters have been applied to the video unit or before the in-loop filters have been applied to the video unit. The in-loop filters may be, for example, the DBF, the SAO, and/or the ALF in FIGS. 6A-6D. In an embodiment, the luma information comprises luma samples of reconstruction, and wherein the luma samples of reconstruction are used as an input to the SR process after a first one of a plurality of in-loop filters have been applied to the video unit but before a last one of the plurality of in-loop filters have been applied to the video unit.

[00458] In an embodiment, an input to a neural network (NN) in the SR process includes both chroma samples of reconstruction and luma samples of reconstruction.

[00459] In an embodiment, the luma samples of reconstruction are down-sampled to a same resolution as the chroma samples of reconstruction and are concatenated with the chroma samples of reconstruction.

[00460] In an embodiment, the luma samples of reconstruction are down-sampled using bi-linear interpolation. By way of example, bi-linear interpolation is a method for interpolating functions of two variables (e.g., x and y) using repeated linear interpolation.

[00461] In an embodiment, the luma samples of reconstruction are down-sampled using bi-cubic interpolation. By way of example, bi-cubic interpolation is an extension of cubic interpolation for interpolating data points on a two-dimensional regular grid. The interpolated surface is smoother than corresponding surfaces obtained by bi-linear interpolation or nearest-neighbor interpolation. Bi-cubic interpolation can be accomplished using either Lagrange polynomials, cubic splines, or cubic convolution algorithm.

[00462] In an embodiment, the luma samples of reconstruction are down-sampled using convolution with a stride equal to a scaling ratio of an original frame. In an embodiment, the luma samples of reconstruction are down-sampled using an inverse of a pixel shuffle.

[00463] In an embodiment, the luma samples of reconstruction are down-sampled based on a color format of 4:2:0 or 4:2:2. Color formats (such as 4:4:4, 4:2:2, and 4:2:0) are discussed.

[00464] For 4:4:4: chroma subsampling, each of the three Y'CbCr components have the same sample rate, thus there is no chroma subsampling. This scheme is sometimes used in high-end film scanners and cinematic post production. YCbCr is a family of color spaces used as a part of the color image pipeline in video and digital photography systems. Y is the luma component and Cb and Cr are the blue-difference and red-difference chroma components.

[00465] For 4:2:2 chroma subsampling, the two chroma components are sampled at half the sample rate of luma: the horizontal chroma resolution is halved. This reduces the bandwidth of an uncompressed video signal by one-third with little to no visual difference.

[00466] For 4:2:0 chroma subsampling, the horizontal sampling is doubled compared to 4:1:1, but as the Cb and Cr channels are only sampled on each alternate line in this scheme, the vertical resolution is halved. The data rate is thus the same. Cb and Cr are each subsampled at a factor of two both horizontally and vertically. There are three variants of 4:2:0 schemes, having different horizontal and vertical siting.

[00467] In MPEG-2, Cb and Cr are co-sited horizontally. Cb and Cr are sited between pixels in the vertical direction (sited interstitially). In Joint Photographic Experts Group (JPEG)/JPEG File Interchange Format (JFIF), H.261, and MPEG-1, Cb and Cr are sited interstitially, halfway between alternate luma samples. In 4:2:0 DV, Cb and Cr are co-sited in the horizontal direction. In the vertical direction, they are co-sited on alternating lines.

[00468] In an embodiment, a method of down-sampling the luma samples of reconstruction is signaled from an encoder to a decoder. In an embodiment, a determination to apply down-sampling to the luma samples of reconstruction is made based on a color format.

[00469] In an embodiment, a color format is 4:4:4 and the luma samples of reconstruction are not down-sampled.

[00470] In an embodiment, the chroma information comprises chroma samples of reconstruction, and wherein the chroma samples of reconstruction are used as an input to the SR process after in-loop filters have been applied to the video unit or before the in-loop filters have been applied to the video unit.

[00471] In an embodiment, the chroma information comprises chroma samples of reconstruction, and wherein the chroma samples of reconstruction are used as an input to the SR process after a first one of a plurality of in-loop filters have been applied to the video unit but before a last one of the plurality of in-loop filters have been applied to the video unit.

[00472] In an embodiment, an input to a neural network (NN) in the SR process includes both chroma samples of reconstruction and luma prediction samples.

[00473] In an embodiment, the first color component is a chroma component (e.g., U) and the second color component (e.g., V) is a different chroma component.

[00474] In an embodiment, the input includes reconstructed samples and decoded information, and wherein the decoded information comprises one or more of partition information, prediction information, and an intra prediction mode.

[00475] In an embodiment, the intra prediction mode comprises a binary frame, wherein each value in the binary frame indicates whether or not one of the reconstructed samples belongs to a skip coded unit, which is also known as a skip mode. Skip mode is a prediction mode. Skip mode is well suited to code static image regions where the prediction error tends to be very small. When the prediction mode of a coding unit (CU) is skip mode, the coding unit may be referred to as a skip coded unit. In an embodiment, a binary frame is a frame composed of binary pixels. Binary pixels are pixels represented by either a 0 or a 1. For example, when the intensity value of a pixel is smaller than the threshold value, the pixel is represented by a first binary value (e.g., 0). When the intensity value of the pixel is larger than the threshold value, the pixel is represented by a second binary value (e.g., 1).

[00476] In an embodiment, the prediction information is derived by motion compensation for an inter predicted coding unit. Motion compensation is an algorithmic technique used to predict a frame in a video, given the previous and/or future frames by accounting for motion of the camera and/or objects in the video.

[00477] In an embodiment, the coded information comprises prediction information, and wherein the prediction information is used as an input of the SR process applied to the video unit for reconstruction.

[00478] In an embodiment, the prediction information comprises luma information of prediction pictures, and wherein the video unit comprises a luma component.

[00479] In an embodiment, the prediction information comprises luma information of prediction pictures, and wherein the video unit comprises a chroma component.

[00480] In an embodiment, the prediction information comprises chroma information of prediction pictures, and wherein the video unit comprises a chroma component.

[00481] In an embodiment, the prediction information comprises luma and chroma information of prediction pictures used together for reconstruction, and wherein the video unit comprises one or more of a luma component and a chroma component.

[00482] In an embodiment, the prediction information comprises padded samples when a coding mode is a palette mode or a pulse-code modulation (PCM) mode. In an embodiment, palette mode is a coding tool included in the HEVC screen content coding extension (SCC) to

improve the coding efficiency for screen contents such as computer generated video with a substantial amount of text and graphics. It is observed that a local area in screen content typically has a few colors separated by sharp edges. To exploit such characteristics, palette mode represents samples in a block with indexes pointing to the color entries in a palette table. PCM coding is usually a lossless coding process that encodes the individual samples of video data without prediction-based compression.

[00483] In an embodiment, the coded information comprises partition information, and wherein the partition information is used as an input of the SR process applied to the video unit for reconstruction.

[00484] In an embodiment, the video unit comprises a low-resolution frame, and wherein the partition information has a same resolution as the low-resolution frame.

[00485] In an embodiment, the coded information comprises an intra prediction mode, and wherein the intra prediction mode is used as an input of the SR process applied to the video unit for reconstruction. In an embodiment, the intra prediction mode is an intra prediction mode of a current sample obtained by intra prediction or inter prediction.

[00486] In an embodiment, a matrix of the intra prediction mode is constructed as one input of the SR process, wherein the matrix has a same resolution as the video unit when reconstructed, and wherein each sample in the matrix has a value corresponding to the intra prediction mode of a corresponding coding unit (CU).

[00487] In an embodiment, the video unit comprises an intra coded slice (I slice) or an intra coded frame (I frame). In an embodiment, the video unit comprises a uni-directional coded slice (P slice) or a bi-directional coded frame (B frame).

[00488] In an embodiment, the method 1700 may utilize or incorporate one or more of the features or processes of the other methods disclosed herein.

[00489] A listing of solutions preferred by some embodiments is provided next.

[00490] The following solutions show example embodiments of techniques discussed in the present disclosure (e.g., Example 1).

[00491] 1. A method of processing video data, comprising: determining, from a coded information of a video block of a video, a super resolution (SR) process for the video block according to a rule; and performing a conversion between the video and a bitstream of the video based on the determining.

[00492] 2. The method of claim 1, wherein the coded information is used as an input to a neural network (NN) based SR process.

[00493] 3. The method of any of claims 1-2, wherein the coded information includes a partitioning type, or a prediction mode information.

[00494] 4. The method of any of claims 1-3, wherein the rule specifies that coded information of a first color component is used for the determining the SR process for a second color component.

[00495] 5. The method of claim 4, wherein the first color component is a chroma component and the second color component is a luma component.

[00496] 6. The method of claim 4, wherein the first color component is a Cr component and the second color component is a Cb component.

[00497] 7. The method of claim 3, wherein the prediction mode information includes prediction mode from a first color component and wherein the video block is in a second color component of the video.

[00498] 8. The method of claim 7, wherein the first color component is a chroma component and the second color component is a luma component.

[00499] 9. The method of claim 3, wherein the partitioning type is from a low resolution frame that has a lower resolution than a frame containing the video block.

[00500] 10. The method of claim 3, wherein the prediction mode information comprises intra prediction mode information.

[00501] 11. The method of any of claims 1-10, wherein the rule is selectively applied to a video unit according to a type of the video unit.

[00502] 12. The method of claim 11, wherein the video unit is a slice or a picture.

[00503] 13. The method of any of above claims, wherein the SR process is used for upsampling reconstructed samples in B slices or P slices based on a neural network based SR model.

[00504] 14. The method of any of claims 1-13 wherein the video unit comprises a picture or a slice.

[00505] 15. The method of any of claims 1-14, wherein the conversion includes generating the bitstream from the video.

[00506] 16. The method of any of claims 1-14, wherein the conversion includes generating the video from the bitstream.

[00507] 17. A video decoding apparatus comprising a processor configured to implement a method recited in one or more of claims 1 to 16.

[00508] 18. A video encoding apparatus comprising a processor configured to implement a method recited in one or more of claims 1 to 16.

[00509] 19. A computer program product having computer code stored thereon, the code, when executed by a processor, causes the processor to implement a method recited in any of claims 1 to 16.

[00510] 20. A method of video processing comprising generating a bitstream according to a method recited in any one or more of claims 1-16 and storing the bitstream on a computer readable medium.

[00511] 21. A method, an apparatus or a system described in the present document.

[00512] The following documents are incorporated by reference in their entirety:

[1] J. Chen, Y. Ye, S. Kim (editors), "Algorithm description for Versatile Video Coding and Test Model 8 (VTM 8)," JVET-Q2002.

[2] VTM software: https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM.git

[3] W. Shi, J. Caballero, et al., "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network," Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. arXiv:1609.05158

[4] J. Lin, D. Liu, H. Yang, H. Li, "Convolutional Neural Network-Based Block Up-Sampling for HEVC", TCSVT 2019.

[00513] The disclosed and other solutions, examples, embodiments, modules and the functional operations described in this document can be implemented in digital electronic circuitry, or in computer software, firmware, or hardware, including the structures disclosed in this document and their structural equivalents, or in combinations of one or more of them. The disclosed and other embodiments can be implemented as one or more computer program products, i.e., one or more modules of computer program instructions encoded on a computer readable medium for execution by, or to control the operation of, data processing apparatus. The computer readable medium can be a machine-readable storage device, a machine-readable storage substrate, a memory device, a composition of matter effecting a machine-readable propagated signal, or a combination of one or more them. The term "data processing apparatus" encompasses all apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them. A propagated signal is an artificially generated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus.

[00514] A computer program (also known as a program, software, software application, script, or code) can be written in any form of programming language, including compiled or interpreted languages, and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A computer program does not necessarily correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data (e.g., one or more scripts stored in a markup language document), in a single file dedicated to the program in question, or in multiple coordinated files (e.g., files that store one or more modules, sub programs, or portions of code). A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[00515] The processes and logic flows described in this document can be performed by one or more programmable processors executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit).

[00516] Processors suitable for the execution of a computer program include, by way of example, both general and special purpose microprocessors, and any one or more processors of any kind of digital computer. Generally, a processor will receive instructions and data from a read only memory or a random-access memory or both. The essential elements of a computer are a processor for performing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Computer readable media suitable for storing computer program instructions and data include all forms of non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and compact disk read-only memory (CD ROM) and digital versatile disc-read only memory (DVD-ROM) disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[00517] While this patent document contains many specifics, these should not be construed as limitations on the scope of any subject matter or of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular techniques. Certain features that are described in this patent document in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[00518] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. Moreover, the separation of various system components in the embodiments described in this patent document should not be understood as requiring such separation in all embodiments.

[00519] Only a few implementations and examples are described and other implementations, enhancements and variations can be made based on what is described and illustrated in this patent document.

[00520] While this patent document contains many specifics, these should not be construed as limitations on the scope of any subject matter or of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular techniques. Certain features that are described in this patent document in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable sub-combination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a sub-combination or variation of a sub-combination.

[00521] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. Moreover, the separation of various system components in the embodiments described

in this patent document should not be understood as requiring such separation in all embodiments.

[00522] Only a few implementations and examples are described and other implementations, enhancements and variations can be made based on what is described and illustrated in this patent document.

CLAIMS

What is claimed is:

1. A method of processing video data, comprising:

using coded information during application of a super resolution (SR) process to a video unit; and

performing a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information.

2. The method of claim 1, wherein the coded information comprises encoded information.

3. The method of claim 1, wherein the coded information comprises decoded information.

4. The method of any of claims 1-3, wherein the SR process comprises a neural network (NN)-based SR process, and wherein the coded information is used as an input to the NN-based SR process.

5. The method of any of claims 1-4, wherein the coded information is used to determine that the SR process, which is one of a plurality of SR processes available, is applied.

6. The method of any of claims 1-5, wherein the coded information comprises one or more of partition information, prediction information, and an intra prediction mode.

7. The method of any of claims 1-5, wherein an input of the SR process comprises reconstructed samples and decoded information, and wherein the decoded information comprises one or more of partition information, prediction information, and an intra prediction mode.

8. The method of any of claims 1-5, wherein the coded information comprises partition information having a same resolution as a reconstructed frame.

9. The method of claim 8, further comprising deriving sample values in the partition information by averaging reconstructed samples in the video unit.

10. The method of any of claims 1-5, wherein the coded information comprises prediction information, and wherein the prediction information comprises prediction samples from intra prediction, intra block copy (IBC), or inter-prediction.

11. The method of any of claims 1-5, wherein the coded information comprises an intra prediction mode having a same resolution as a reconstructed frame.

12. The method of claim 11, further comprising deriving sample values in the intra prediction mode by filling the intra prediction mode in a corresponding video unit.

13. The method of any of claims 1-5, wherein the coded information comprises quantization parameter (QP) information, and wherein the QP information is used to generate an up-sampled reconstruction.

14. The method of any of claims 1-5, further comprising filling a matrix with quantization (QP) parameter values to construct a QP map, wherein the QP map is used as an input to the SR process, and wherein a spatial resolution of the QP map is the same as the spatial resolution of another input of the SR process.

15. The method of claim 1, wherein the video unit comprises a first color component, and wherein a second color component is utilized during the SR process applied to the first color component.

16. The method of claim 1, wherein the video unit comprises a first color component, and wherein a second color component is used as an input to the SR process applied to the first color component.

17. The method of claim 1, wherein the SR process comprises a luma up-sampling process, and wherein chroma information is used as an input to the luma up-sampling process.

18. The method of claim 1, wherein the SR process comprises a chroma up-sampling process, and wherein luma information is used as an input to the chroma up-sampling process.

19. The method of claim 18, wherein the luma information comprises luma samples of reconstruction, and wherein the luma samples of reconstruction are used as an input to the SR process after in-loop filters have been applied to the video unit or before the in-loop filters have been applied to the video unit.

20. The method of claim 18, wherein the luma information comprises luma samples of reconstruction, and wherein the luma samples of reconstruction are used as an input to the SR process after a first one of a plurality of in-loop filters have been applied to the video unit but before a last one of the plurality of in-loop filters have been applied to the video unit.

21. The method of claim 1, wherein an input to a neural network (NN) in the SR process includes both chroma samples of reconstruction and luma samples of reconstruction.

22. The method of claim 21, wherein the luma samples of reconstruction are down-sampled to a same resolution as the chroma samples of reconstruction and are concatenated with the chroma samples of reconstruction.

23. The method of claim 22, wherein the luma samples of reconstruction are down-sampled using bi-linear interpolation.

24. The method of claim 22, wherein the luma samples of reconstruction are down-sampled using bi-cubic interpolation.

25. The method of claim 22, wherein the luma samples of reconstruction are down-sampled using convolution with a stride equal to a scaling ratio of an original frame.

26. The method of claim 22, wherein the luma samples of reconstruction are down-sampled using an inverse of a pixel shuffle.

27. The method of claim 22, wherein the luma samples of reconstruction are down-sampled based on a color format of 4:2:0 or 4:2:2.

28. The method of any of claims 23-27, wherein a method of down-sampling the luma samples of reconstruction is signaled from an encoder to a decoder.

29. The method of any of claims 23-27, wherein a determination to apply down-sampling to the luma samples of reconstruction is made based on a color format.

30. The method of claim 21, wherein a color format is 4:4:4 and the luma samples of reconstruction are not down-sampled.

31. The method of claim 17, wherein the chroma information comprises chroma samples of reconstruction, and wherein the chroma samples of reconstruction are used as an input to the SR process after in-loop filters have been applied to the video unit or before the in-loop filters have been applied to the video unit.

32. The method of claim 17, wherein the chroma information comprises chroma samples of reconstruction, and wherein the chroma samples of reconstruction are used as an input to the SR process after a first one of a plurality of in-loop filters have been applied to the video unit but before a last one of the plurality of in-loop filters have been applied to the video unit.

33. The method of claim 1, wherein an input to a neural network (NN) in the SR process includes both chroma samples of reconstruction and luma prediction samples.

34. The method of claim 15, wherein the first color component is a chroma component and the second color component is a different chroma component.

35. The method of claim 16, wherein the input includes reconstructed samples and decoded information, and wherein the decoded information comprises one or more of partition information, prediction information, and an intra prediction mode.

36. The method of claim 35, wherein the intra prediction mode comprises a binary frame, wherein each value in the binary frame indicates whether or not one of the reconstructed samples belongs to a skip mode.

37. The method of claim 35, wherein the prediction information is derived by motion compensation for an inter predicted coding unit.

38. The method of claim 1, wherein the coded information comprises prediction information, and wherein the prediction information is used as an input of the SR process applied to the video unit for reconstruction.

39. The method of claim 38, wherein the prediction information comprises luma information of prediction pictures, and wherein the video unit comprises a luma component.

40. The method of claim 38, wherein the prediction information comprises luma information of prediction pictures, and wherein the video unit comprises a chroma component.

41. The method of claim 38, wherein the prediction information comprises chroma information of prediction pictures, and wherein the video unit comprises a chroma component.

42. The method of claim 38, wherein the prediction information comprises luma and chroma information of prediction pictures used together for reconstruction, and wherein the video unit comprises one or more of a luma component and a chroma component.

43. The method of claim 38, wherein the prediction information comprises padded samples when a coding mode is a palette mode or a pulse-code modulation mode.

44. The method of claim 1, wherein the coded information comprises partition information, and wherein the partition information is used as an input of the SR process applied to the video unit for reconstruction.

45. The method of claim 44, wherein the video unit comprises a frame, and wherein the partition information has a same resolution as the frame.

46. The method of claim 1, wherein the coded information comprises an intra prediction mode, and wherein the intra prediction mode is used as an input of the SR process applied to the video unit for reconstruction.

47. The method of claim 46, wherein the intra prediction mode is an intra prediction mode of a current sample obtained by intra prediction or inter prediction.

48. The method of claim 47, wherein a matrix of the intra prediction mode is constructed as one input of the SR process, wherein the matrix has a same resolution as the video unit when reconstructed, and wherein each sample in the matrix has a value corresponding to the intra prediction mode of a corresponding coding unit (CU).

49. The method of claim 1, wherein the video unit comprises an intra coded slice (I slice) or an intra coded frame (I frame).

50. The method of claim 1, wherein the video unit comprises a uni-directional coded slice (P slice) or a bi-directional coded frame (B frame).

51. The method of claim 1, wherein the conversion includes encoding the video data into the bitstream.

52. The method of claim 1, wherein the conversion includes decoding the video data from the bitstream.

53. An apparatus for processing media data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to:

use coded information during application of a super resolution (SR) process to a video unit; and

perform a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information.

54. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises:

using coded information during application of a super resolution (SR) process to a video unit; and

generating the bitstream based on the SR process as applied using the coded information.

55. An apparatus for processing media data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to perform a method recited in one or more of claims 1 to 52.

56. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method, recited in one or more of claims 1 to 52, performed by a video processing apparatus.

57. A computer-readable program medium having code stored thereupon, the code comprising instructions that, when executed by a processor, causes the processor to implement a method recited in one or more of claims 1 to 52.

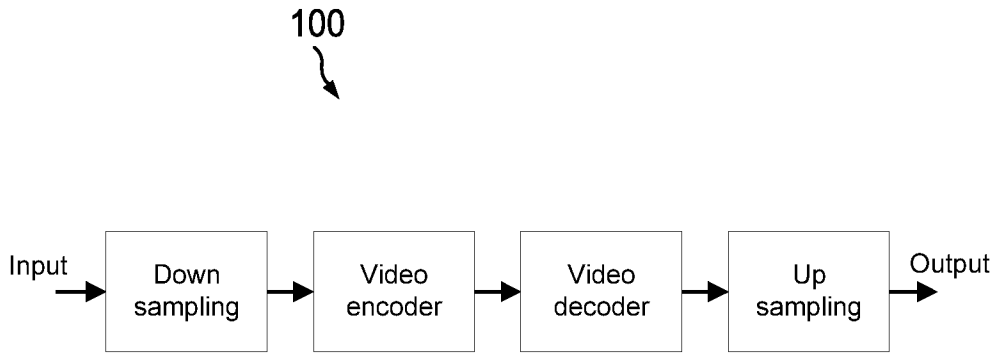

100
Input
Down sampling
Video encoder
Video decoder
Up sampling
Output

FIG. 1

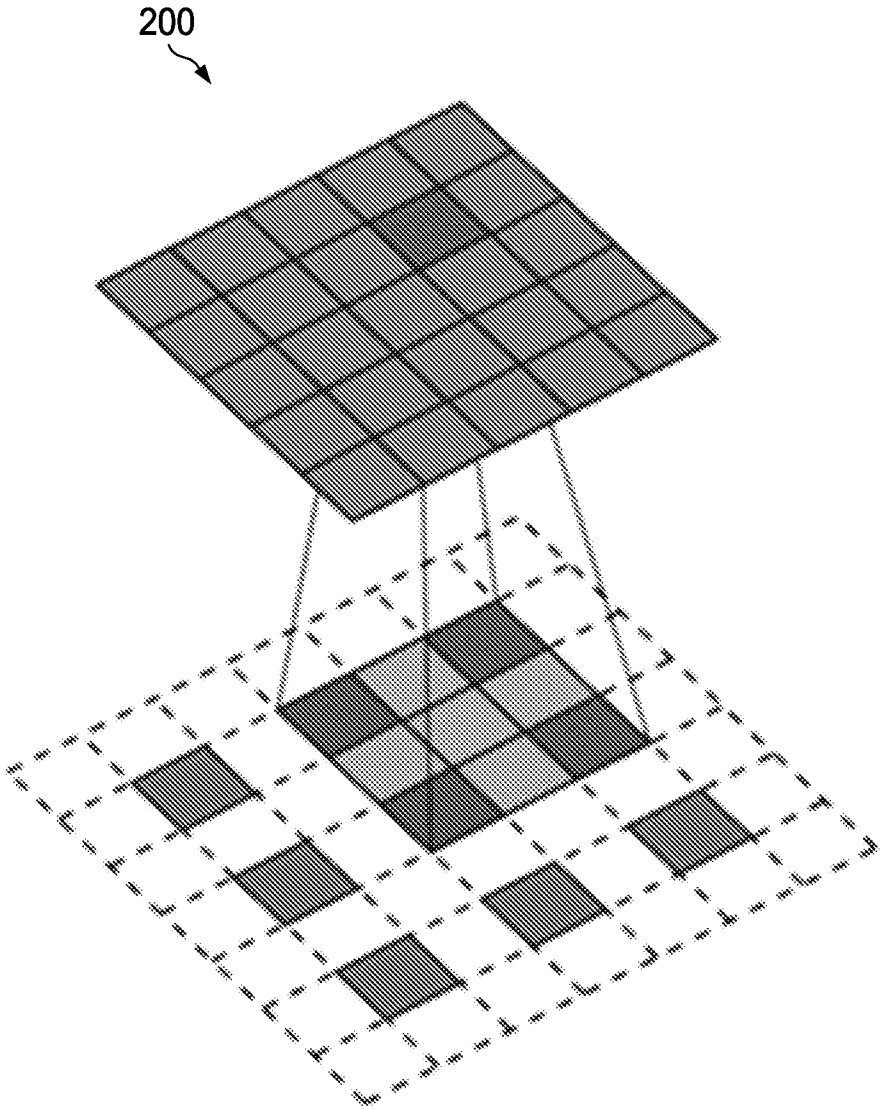


FIG. 2

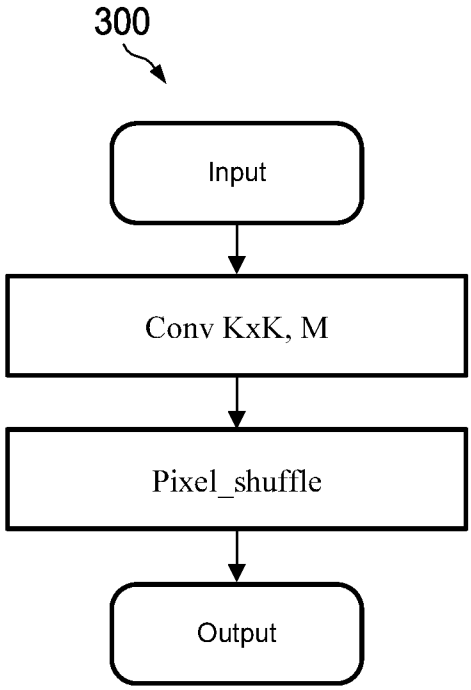

300
Input
Conv KxK, M
Pixel_shuffle
Output

FIG. 3

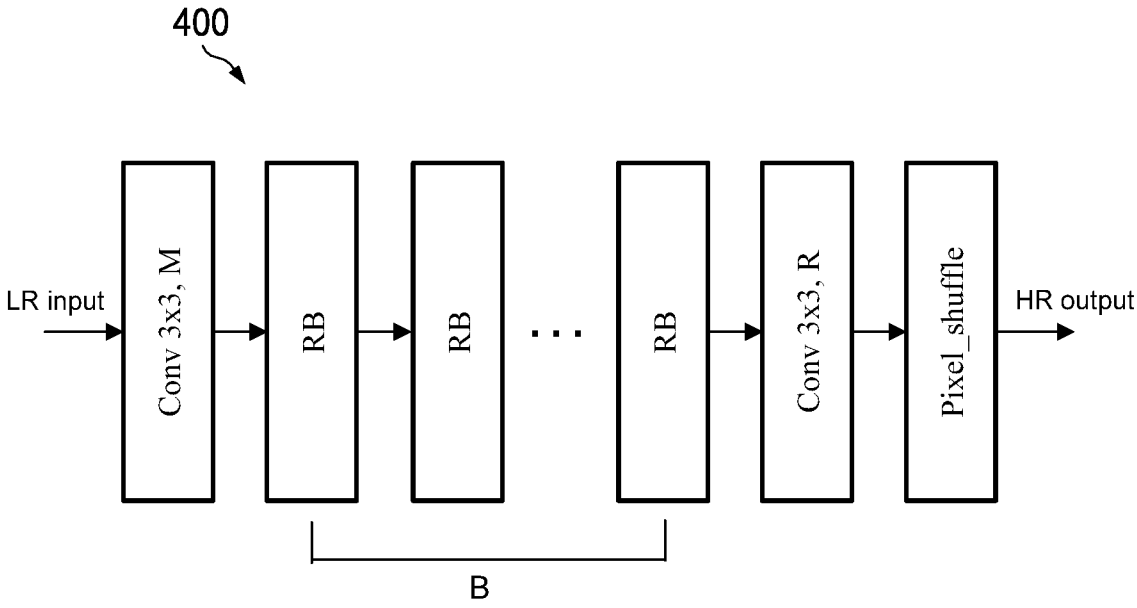

400
LR input
Conv 3x3, M
RB
RB
...
RB
Conv 3x3, R
Pixel_shuffle
HR output
B

FIG. 4

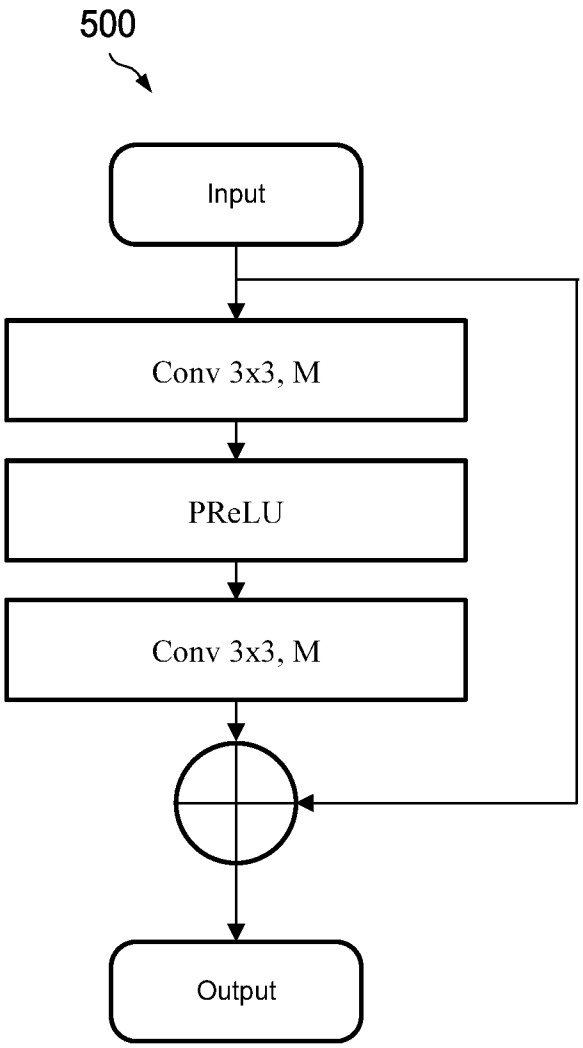


FIG. 5

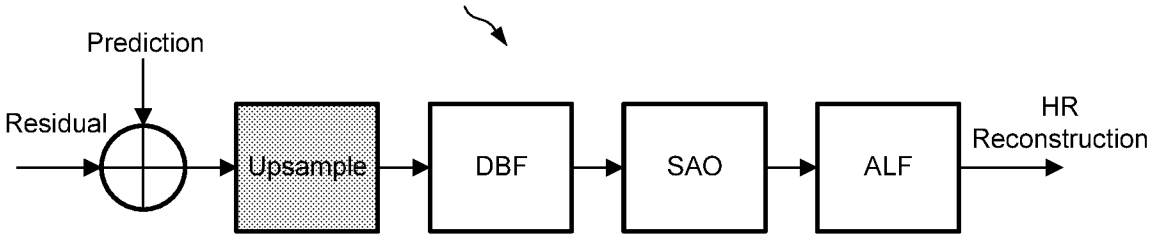


FIG. 6A

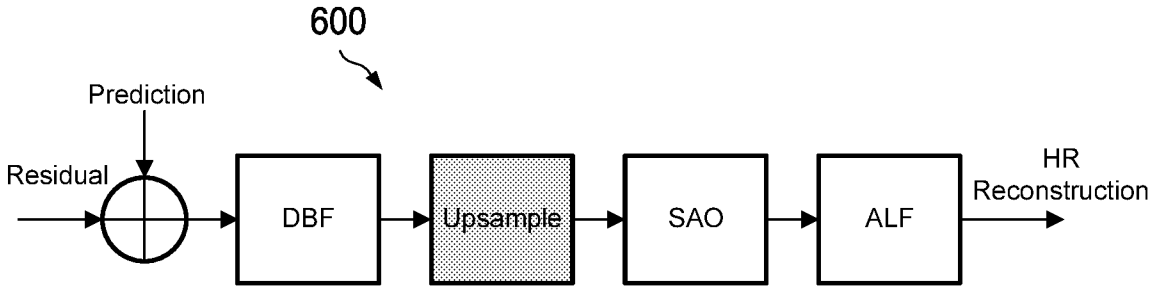


FIG. 6B

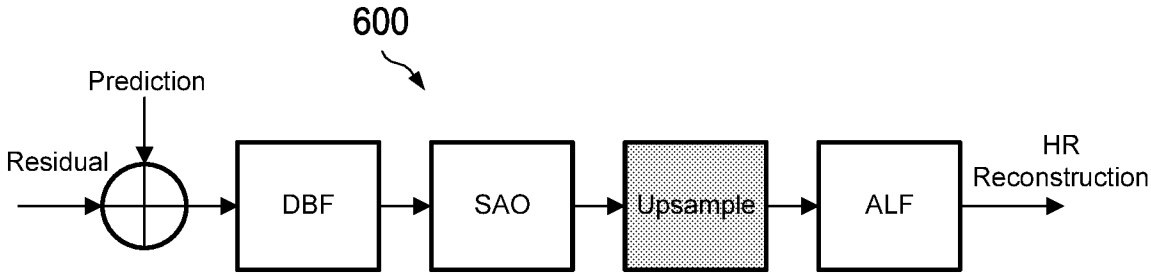


FIG. 6C

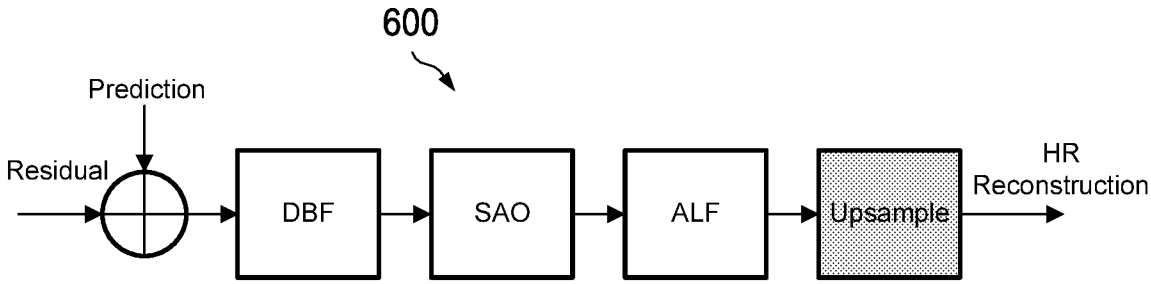


FIG. 6D

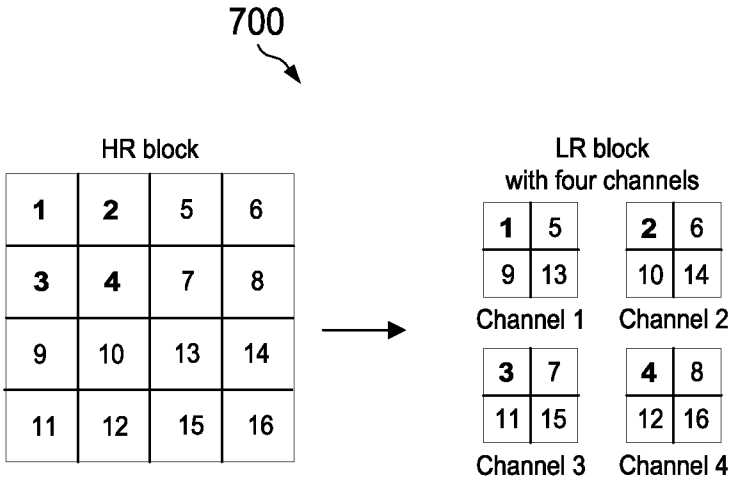

700
HR block
1 2 5 6
3 4 7 8
9 10 13 14
11 12 15 16
LR block
with four channels
1 5
9 13
Channel 1
2 6
10 14
Channel 2
3 7
11 15
Channel 3
4 8
12 16
Channel 4

FIG. 7

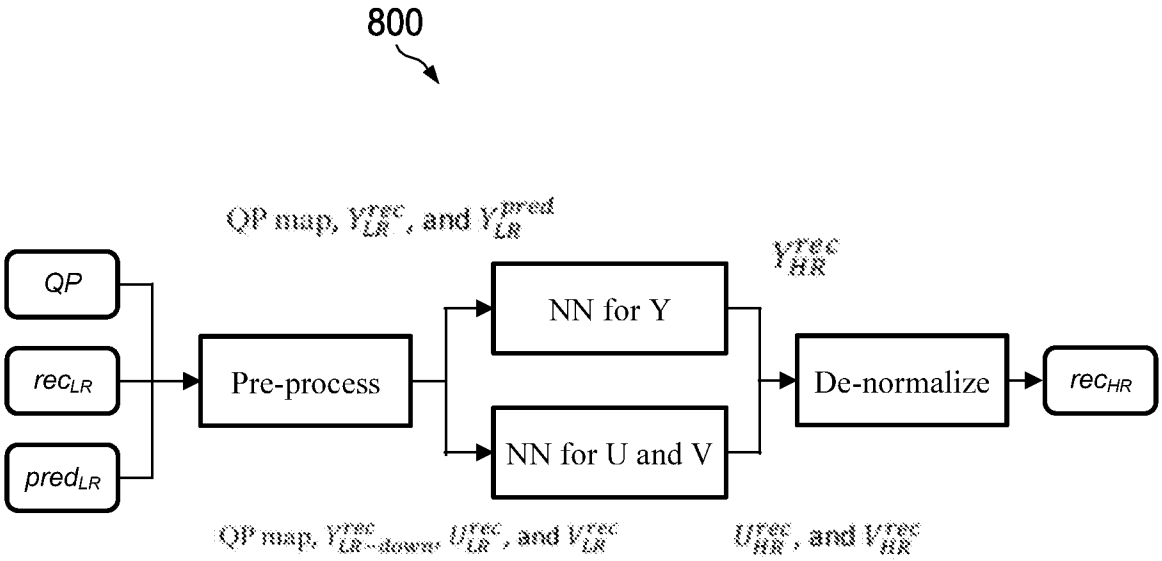

800
QP map, Y_{LR}^{rec}, and Y_{LR}^{pred}
Y_{HR}^{rec}
QP
rec_{LR}
$pred_{LR}$
Pre-process
NN for Y
NN for U and V
De-normalize
rec_{HR}
QP map, $Y_{LR-down}^{rec}$, U_{LR}^{rec}, and V_{LR}^{rec}
U_{HR}^{rec}, and V_{HR}^{rec}

FIG. 8

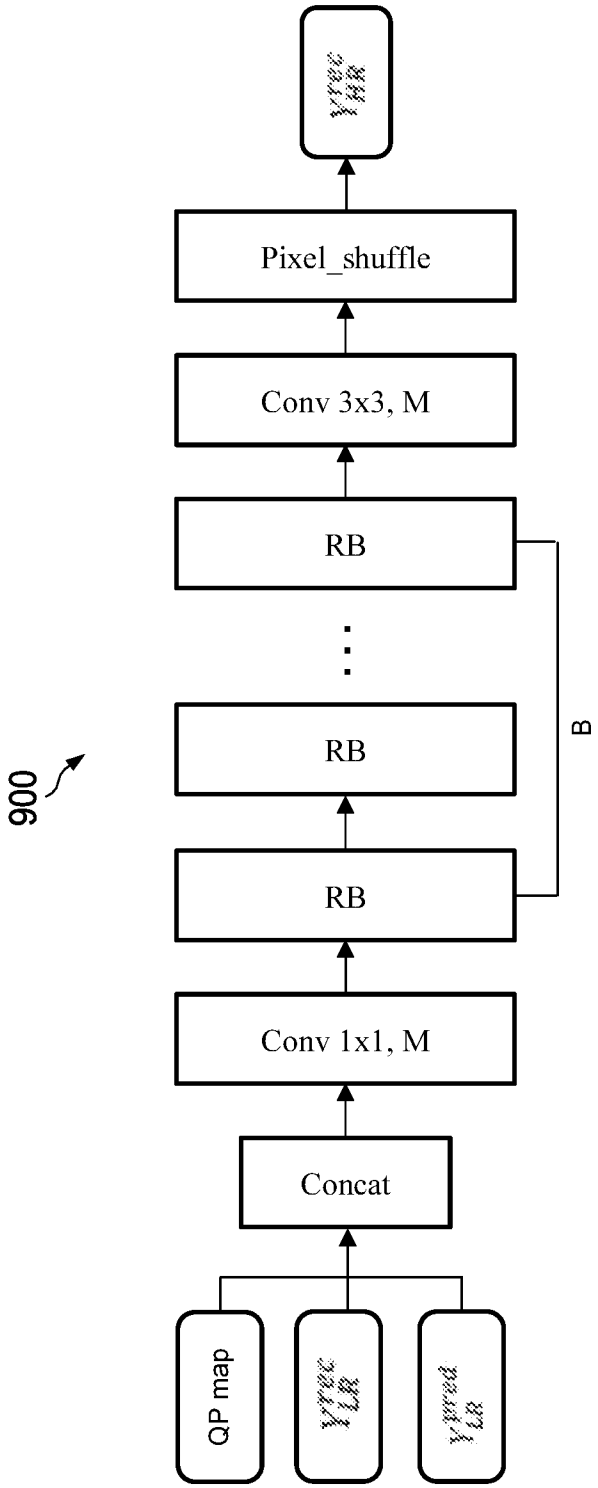


FIG. 9

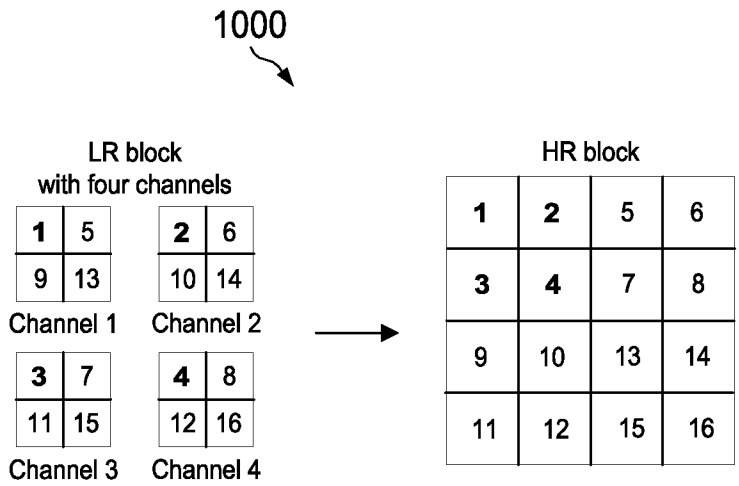

1000
LR block
with four channels
1 5
9 13
Channel 1
2 6
10 14
Channel 2
3 7
11 15
Channel 3
4 8
12 16
Channel 4
HR block
1 2 5 6
3 4 7 8
9 10 13 14
11 12 15 16

FIG. 10

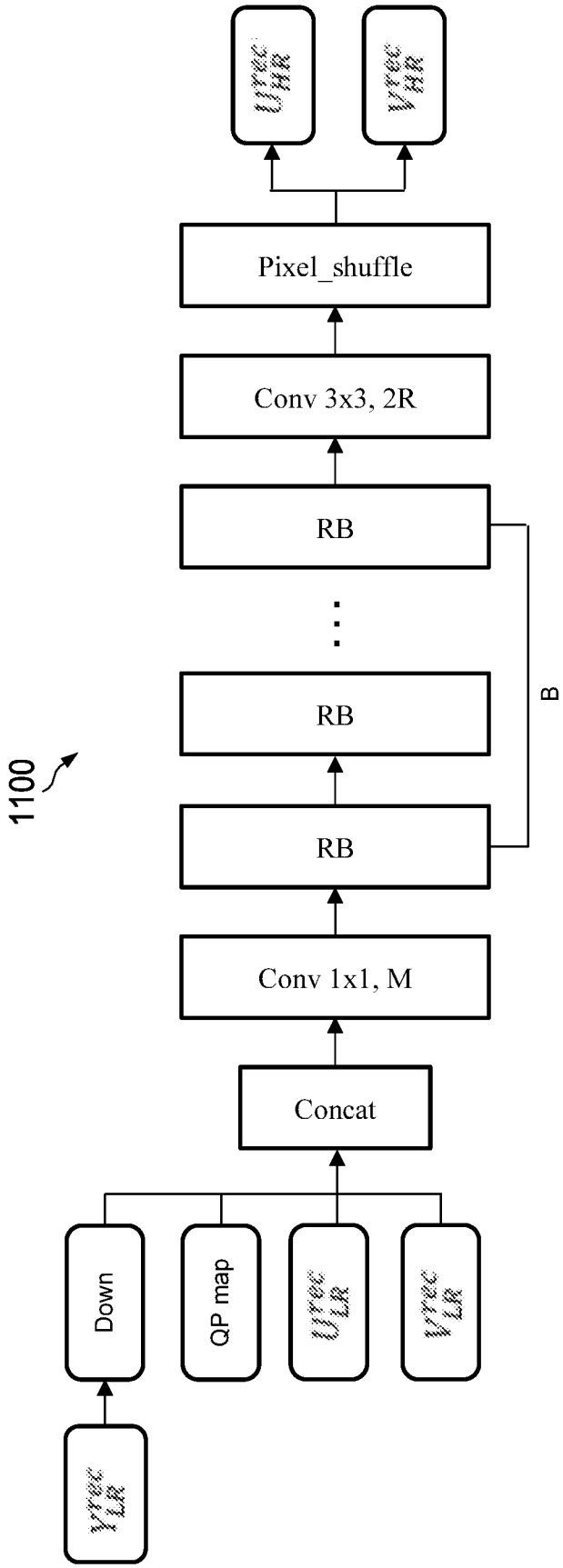

1100
Down
QP map
Concat
Conv 1x1, M
RB
RB
RB
B
Conv 3x3, 2R
Pixel_shuffle

FIG. 11

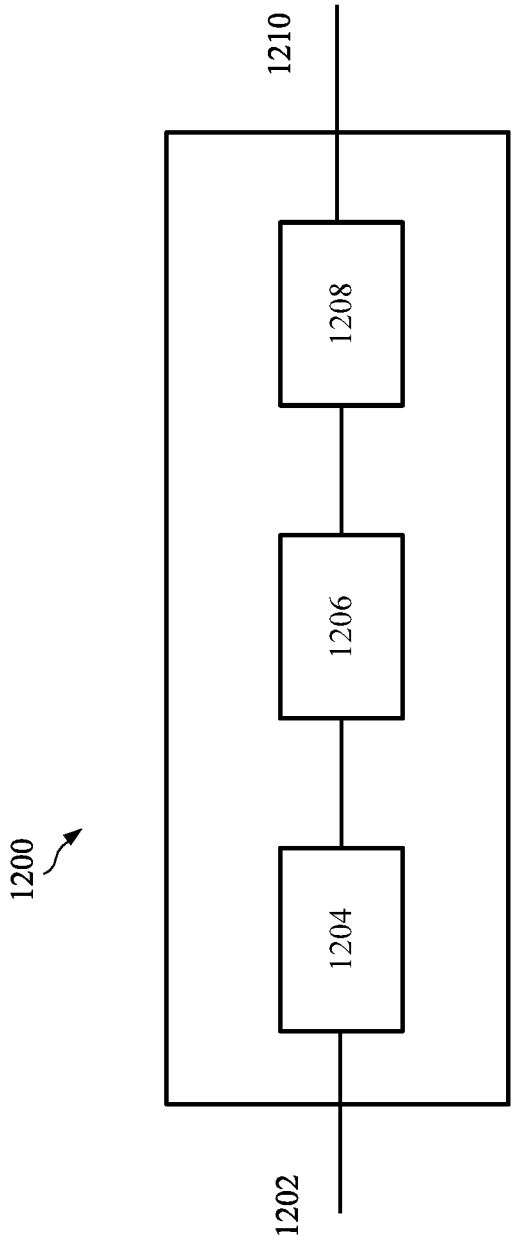


FIG. 12

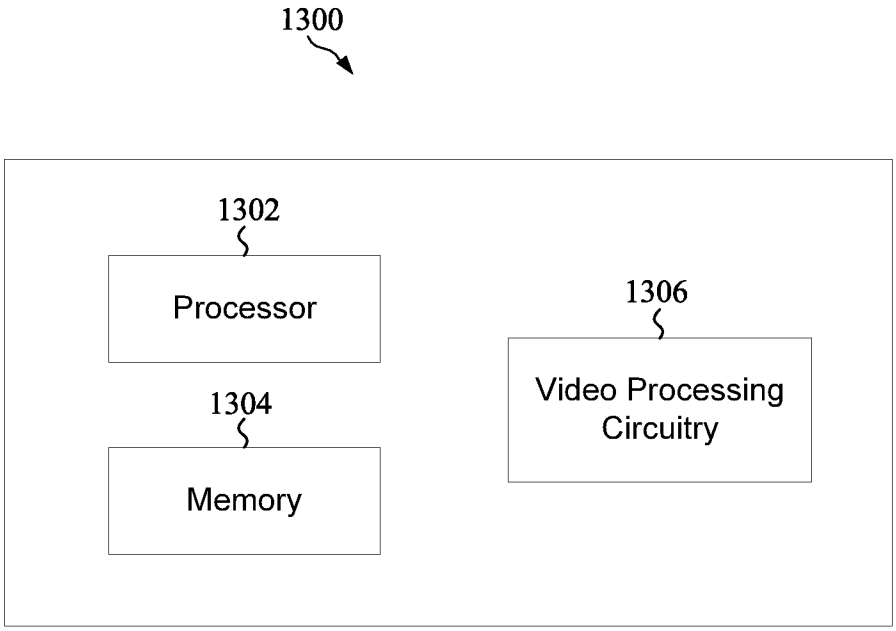

1300
1302
Processor
1304
Memory
1306
Video Processing Circuitry

FIG. 13

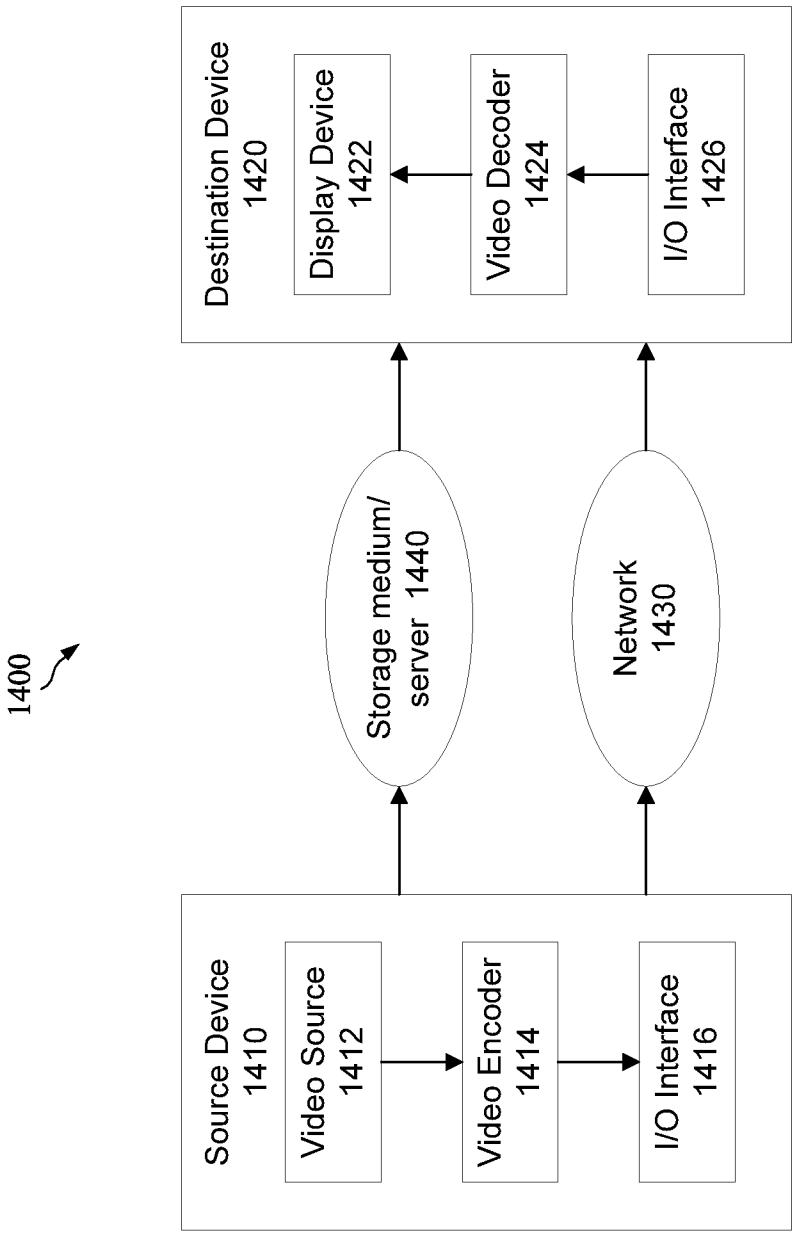

1400
Source Device 1410
Video Source 1412
Video Encoder 1414
I/O Interface 1416
Storage medium/ server 1440
Network 1430
Destination Device 1420
Display Device 1422
Video Decoder 1424
I/O Interface 1426

FIG. 14

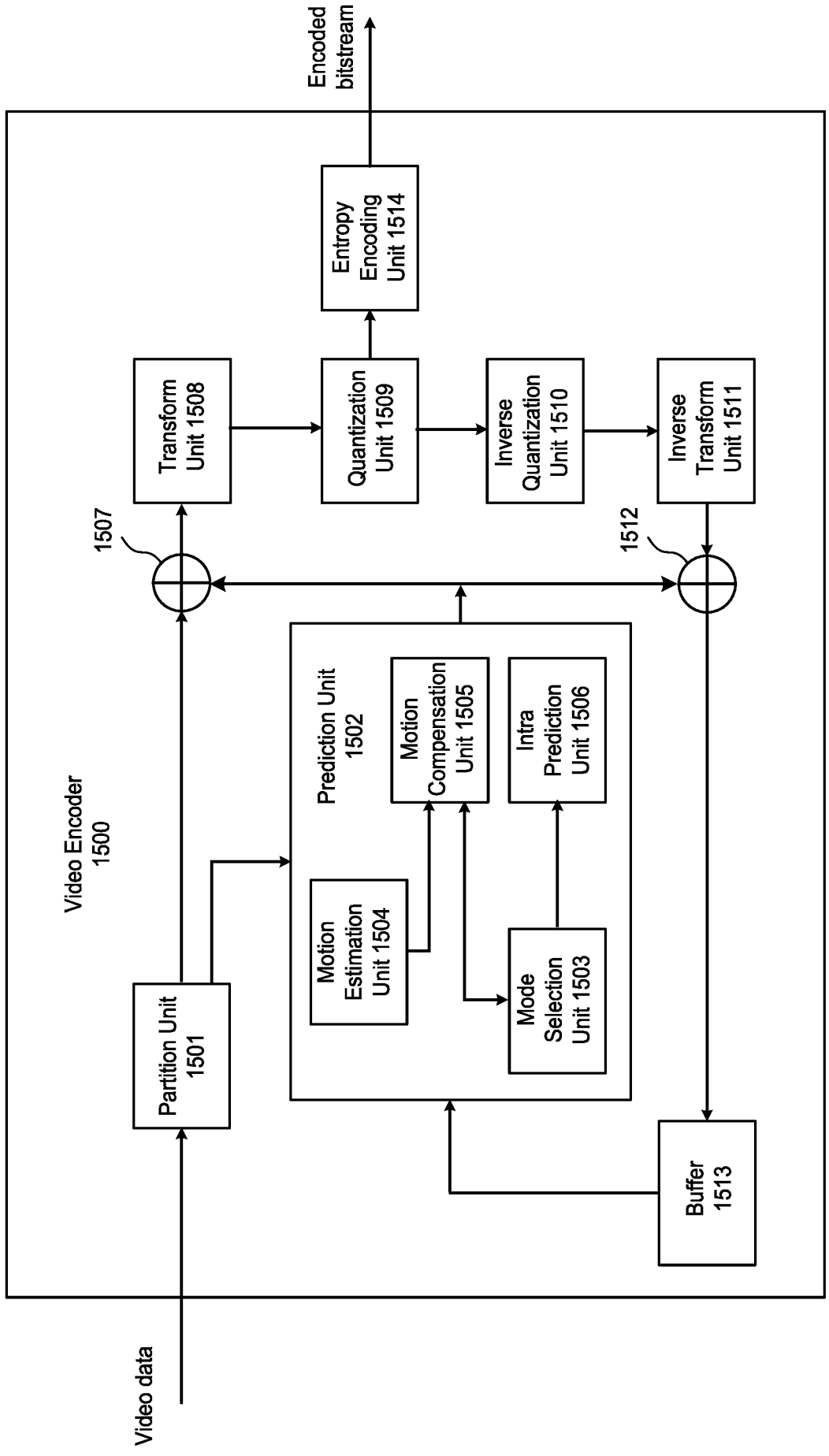

Video data
Video Encoder
1500
Partition Unit
1501
Prediction Unit
1502
Motion
Estimation
Unit 1504
Motion
Compensation
Unit 1505
Intra
Prediction
Unit 1506
Mode
Selection
Unit 1503
1507
Transform
Unit 1508
Quantization
Unit 1509
Entropy
Encoding
Unit 1514
Encoded
bitstream
Inverse
Quantization
Unit 1510
Inverse
Transform
Unit 1511
1512
Buffer
1513

FIG. 15

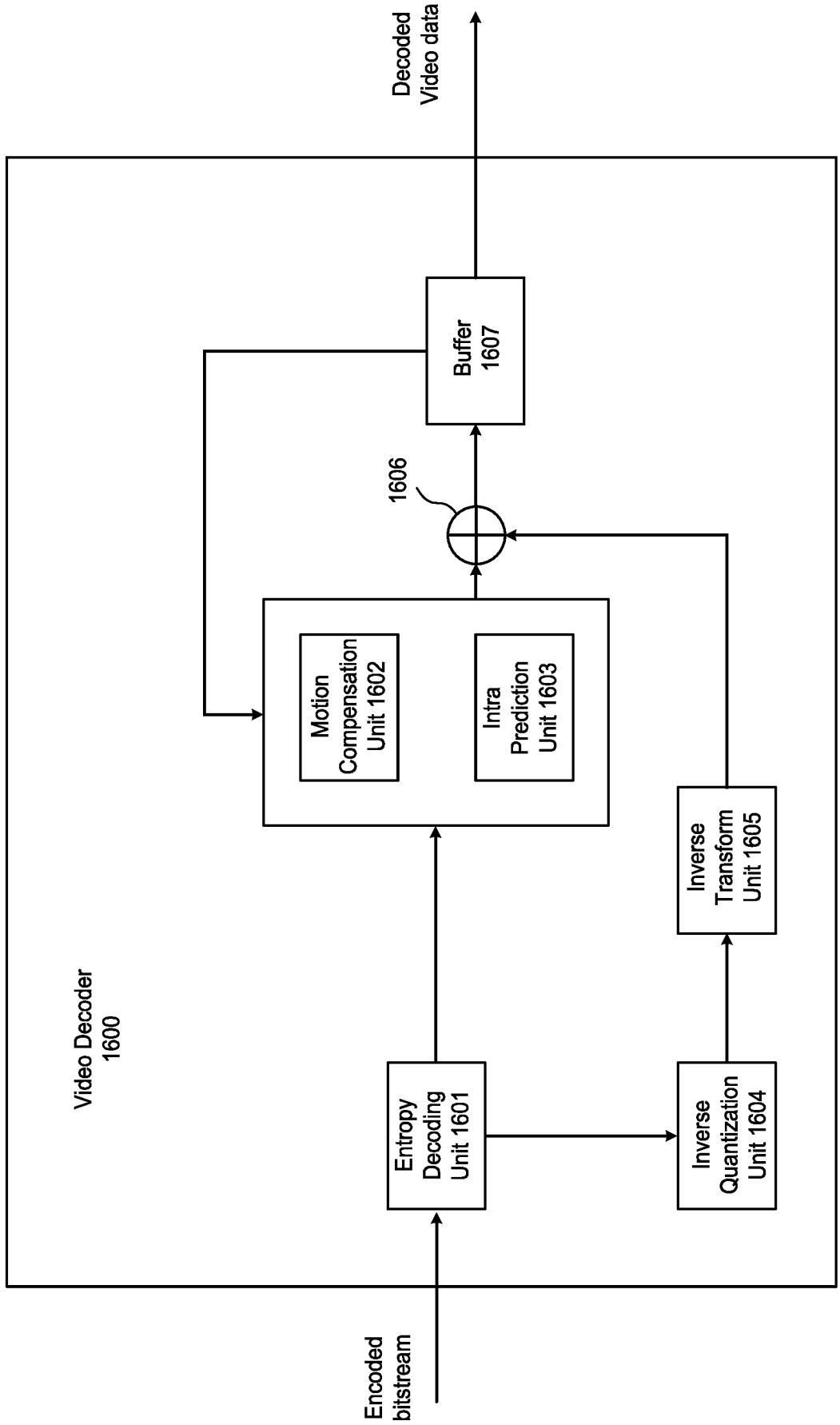

Decoded Video data
Buffer 1607
1606
Motion Compensation Unit 1602
Intra Prediction Unit 1603
Inverse Transform Unit 1605
Video Decoder 1600
Entropy Decoding Unit 1601
Inverse Quantization Unit 1604
Encoded bitstream

FIG. 16

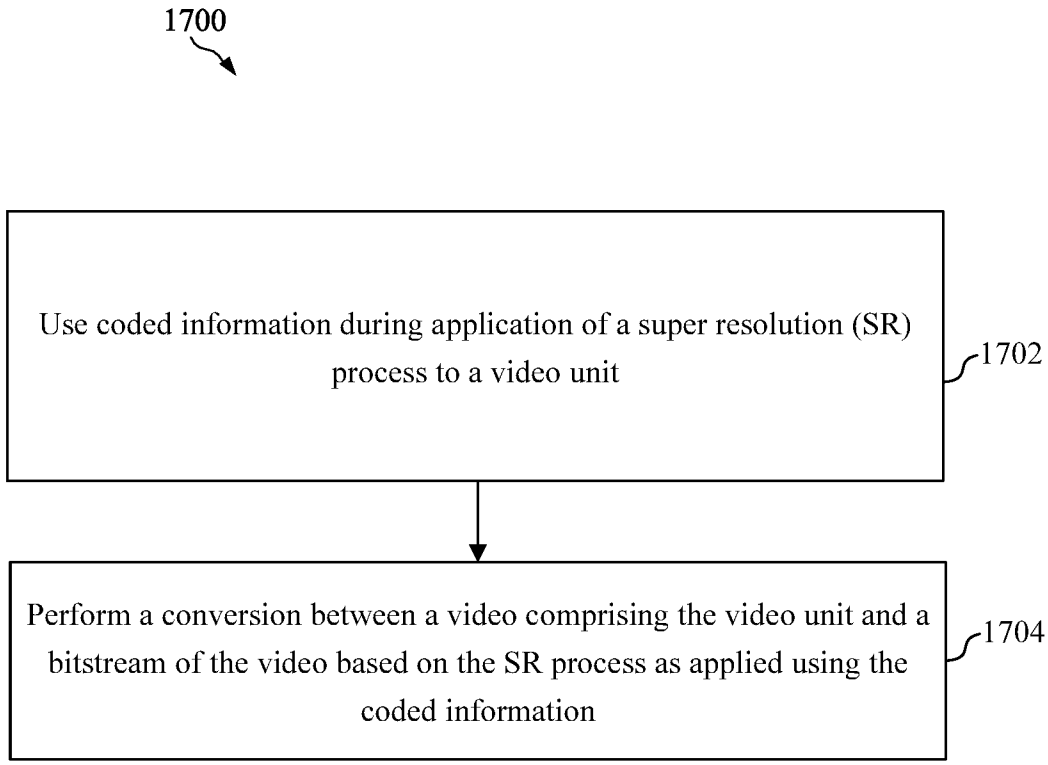

1700
Use coded information during application of a super resolution (SR) process to a video unit
1702
Perform a conversion between a video comprising the video unit and a bitstream of the video based on the SR process as applied using the coded information
1704

FIG. 17

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2022/103238

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/59(2014.01)i; G06T 3/40(2006.01)i; G06N 3/04(2006.01)i

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; G06N

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)

CNPAT,WPI,EPODOC,IEEE,CNKI,JVET:up-sampl+, super resolution, neural network, NN, CNN, side information, auxiliary information, video, encod+, decod+, predict+, partition, quantization, QP

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 110136057 A (HANGZHOU HIKVISION DIGITAL TECHNOLOGY CO., LTD.) 16 August 2019 (2019-08-16) specification paragraphs 0118-0119, 0152-0178, 0241-0244, Figures 2-4	1-9, 11-14, 44-57
Y	CN 110136057 A (HANGZHOU HIKVISION DIGITAL TECHNOLOGY CO., LTD.) 16 August 2019 (2019-08-16) specification paragraphs 0118-0119, 0152-0178, 0241-0244, Figures 2-4	6, 7, 10, 15-43
Y	LIN, Jianping et al. "Convolutional Neural Network-Based Block Up-Sampling for HEVC" *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY,* Vol. 29, No. 12, 31 December 2019 (2019-12-31), part IV	6, 7, 10, 33, 35, 37-43
Y	LI, Yue et al. "Convolutional Neural Network-Based Block Up-Sampling for Intra Frame Coding" *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY,* Vol. 28, No. 9, 30 September 2018 (2018-09-30), abstract, part IV	15-32, 34-37
A	CN 111510739 A (HUAWEI TECHNOLOGIES CO., LTD.) 07 August 2020 (2020-08-07) the whole document	1-57

☑ Further documents are listed in the continuation of Box C. ☑ 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
08 September 2022	**28 September 2022**
Name and mailing address of the ISA/CN **National Intellectual Property Administration, PRC** **6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088, China** Facsimile No. **(86-10)62019451**	Authorized officer **RONG,Fang** Telephone No. **86-10-53961814**

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2018137603 A1 (UMBO CV INC.) 17 May 2018 (2018-05-17) the whole document	1-57
A	CHUJOH, Takeshi et al. "AHG9/AHG11 Level information for super-resolution neural network" *Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29 21st Meeting, by teleconference, 6–15 Jan. 2021*, 15 January 2021 (2021-01-15), the whole document	1-57
A	US 2021118095 A1 (SAMSUNG ELECTRONICS CO., LTD.) 22 April 2021 (2021-04-22) the whole document	1-57

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2022/103238

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 110136057 A	16 August 2019	None	
CN 111510739 A	07 August 2020	None	
US 2018137603 A1	17 May 2018	TW 201818346 A	16 May 2018
		EP 3319039 A1	09 May 2018
US 2021118095 A1	22 April 2021	KR 20210045828 A	27 April 2021