

CONTEXT-BASED IMAGE CODING

BACKGROUND

In this document, “coding” may include encoding and/or decoding. Usually, a frame of a video is encoded by an encoder at a transmitting terminal to compress the frame of the video for transmission over a network. The encoding of a given frame may be performed by referring to another frame in a reference video. A bitstream corresponding to an encoded representation generated through the encoding is transmitted to a receiving terminal. A corresponding decoder at the receiving terminal may decode the given frame of the video from the received bitstream, so as to output the decoded given frame to a screen of the receiving terminal. During the coding, the reconstruction quality and the compression efficiency of the frame are always an aspect which is noteworthy.

SUMMARY

According to implementations of the present disclosure, there is provided a context-based image coding solution. In the solution, a reference image of a target image is obtained. A contextual feature representation is extracted from the reference image, the contextual feature representation characterizing contextual information associated with the target image. Conditional encoding or conditional decoding is performed on the target image based on the contextual feature representation. In this way, the enhancement of the performance is achieved in terms of the reconstruction quality and the compression efficiency.

The Summary is to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. The Summary is not intended to identify key features or essential features of the subject matter described herein, nor is it intended to be used to limit the scope of the subject matter described herein.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 illustrates a schematic block diagram of a conventional residual-based video coding system; Fig. 2 illustrates a schematic block diagram of a context-based video coding system according to some implementations of the present disclosure; Fig. 3 illustrates an example of a capability of a contextual feature representation characterizing contextual information according to some implementations of the present disclosure; Fig. 4 illustrates a block diagram of an example structure of a context generator in the system of Fig. 2 according to some implementations of the present disclosure; Fig. 5 illustrates a block diagram of an example structure of an entropy model in the system of Fig. 2 according to some implementations of the present disclosure; Fig. 6 illustrates a comparison between a context-based video coding solution according to some

implementations of the present disclosure and a conventional video coding solution;

Fig. 7 illustrates a flowchart of a video coding process according to some implementations of the present disclosure; and

Fig. 8 illustrates a block diagram of a computing device capable of implementing a plurality of
5 implementations of the present disclosure.

Throughout the drawings, the same or similar reference symbols refer to the same or similar elements.

DETAILED DESCRIPTION OF EMBODIMENTS

The present disclosure will now be described with reference to some example implementations. It
10 is to be understood that these implementations are described only for the purpose of illustration and help those skilled in the art to better understand and thus implement the present disclosure, without suggesting any limitations to the scope of the present disclosure.

As used herein, the term “includes” and its variants are to be read as open terms that mean “includes, but is not limited to.” The term “based on” is to be read as “based at least in part on.”

15 The terms “an implementation” and “one implementation” are to be read as “at least one implementation.” The term “another implementation” is to be read as “at least one other implementation.” The term “first,” “second,” and the like may refer to different or the same objects. Other definitions, either explicit or implicit, may be included below.

As used herein, the term “model” may refer to an association between corresponding input and
20 output learnable from the training data, and thus a corresponding output may be generated for a given input after the training. The generation of the model may be based on machine learning techniques. Deep learning is one of machine learning algorithms that processes the input and provides the corresponding output using a plurality of layers of processing units. A neural network model is an example of a deep learning-based model. As used herein, “model” may also be referred
25 to as “machine learning model”, “learning model”, “machine learning network” or “learning network”, which terms are used interchangeably herein.

A “neural network” is a machine learning network based on deep learning. The neural network can process an input to provide a corresponding output, and usually includes an input layer, an output layer, and one or more hidden layers between the input layer and the output layer. The
30 neural network used in deep learning applications usually includes a large number of hidden layers, thereby increasing the depth of the network. The layers of the neural network are connected in order, so that the output of a preceding layer is provided as the input of a next layer, where the input layer receives the input of the neural network, and the output of the output layer is regarded as a final output of the neural network. Each layer of the neural network includes one or more
35 nodes (also referred to as processing nodes or neurons), each of which processes input from the

preceding layer.

Generally, machine learning may include three phases, i.e., a training phase, a test phase, and an application phase (also referred to as an interference phase). In the training phase, a given model may be trained by using a great amount of training data, with parameter values being iteratively updated until the model can obtain, from the training data, consistent interference that meets an expected target. Through the training, the model may be considered as being capable of learning the association between input and the output (also referred to as input-to-output mapping) from the training data. The parameter values of the trained model are determined. In the test phase, a test input is applied to the trained model to test whether the model can provide a correct output, so as to determine the performance of the model. In the application phase, the model may be used to process an actual input based on the parameter values obtained in the training and to determine the corresponding output.

In this document, a “frame” or “video frame” refers to individual images in a video segment. “Image” and “frame” are used interchangeably in this document. A plurality of consecutive images may form a dynamic video segment, where each image is considered as a frame.

Currently, with the development of the machine learning technology, it has been proposed to apply machine learning to a video coding process. However, the reconstruction quality and compression efficiency of video frames still need to be improved due to limitations of the conventional coding process.

Residual-based conventional video coding

Conventional video coding solutions, including the H.261 video coding standard developed in 1988 to the H.266 video coding standard released in 2020, all widely employ a residual-based coding solution. The solution is based on a predictive coding paradigm by generating a reference image of the current image, and performing encoding and decoding on a residual between the current image and the reference image. Fig. 1 shows a schematic block diagram of a conventional residual-based video coding system 100. The system 100 includes an encoder 110, a decoder 120, an image predictor 130, a residual generation module 140 and a residual addition module 150. In the residual-based coding process, the encoder 110 is referred to as a residual encoder, and the decoder 120 is referred to as a residual decoder.

Assuming that an image 102 to be encoded currently is an image x_t at time t in a video segment, the image predictor 130 is configured to generate a predicted image $\hat{x}_t$ 132 for the image 102 based on the reference image 170. The reference image 170 may include a decoded image $\hat{x}_{t-1}$ at time $t-1$ before t in the video segment. The residual generation module 140 calculates the residual between the image 102 x_t and the predicted image $\hat{x}_t$ 132. The encoder 110 encodes

the residual to generate an encoded representation of the image 102 x_t . A bitstream 112 corresponding to the encoded representation is transmitted to the decoding side.

On the decoding side, the decoder 120 receives the bitstream 112, and decodes the bitstream 112 to get an decoded image. The residual addition module 150 adds up the decoded image provided

- 5 by the decoder 120 and the predicted image $\tilde{x}_t$ 132 generated by the image predictor 130 to obtain the decoded image 160 $\hat{x}_t$ at time t.

The residual-based video coding may be represented as follows:

$$\hat{x}_t = f_{dec}(\lfloor f_{enc}(x_t - \tilde{x}_t) \rfloor) + \tilde{x}_t$$

where $\tilde{x}_t = f_{predict}(\hat{x}_{t-1})$ (1)

- 10 In the above Equation (1), $f_{enc}(\cdot)$ represents an encoding process of the encoder 110, $f_{dec}(\cdot)$ represents a decoding process of the decoder 120, $f_{predict}(\cdot)$ represents a prediction process of the image predictor 130, and $\lfloor \cdot \rfloor$ represents a quantization operation. In an application based on machine learning, the encoder 110 may use a machine learning model to implement residual encoding, and accordingly, the decoder 120 may use the machine learning model to implement residual decoding.

Working principle and example system

Considering the strong temporal correlation between frames in a video, residual encoding was considered as a simple and effective way of compressing the video in the past. However, the Inventor of the present application discovers through research that the residual coding by encoding

- 20 the current image x_t with the predicted image $\hat{x}_t$ being given is not optimal because the residual coding always removes redundancy between the images by using a simple subtraction operation. An entropy of the residual coding is greater than or equal to an entropy of the conditional coding $H(x_t - \tilde{x}_t) \geq H(x_t | \tilde{x}_t)$, where H is a Shannon entropy. Theoretically, a pixel of the current image x_t is related to all pixels in the decoded image at a previous time, and these pixels have been decoded in the image x_t . For a traditional codec, it is difficult to explicitly characterize all the correlations between the decoded image at the previous time and the current image by handcrafted rules. Therefore, the residual-based coding makes use of an assumption that the pixels of the current image are only related to corresponding predicted pixels in the predicted image, thereby simplifying the coding process. However, such a coding solution
- 25 is indeed not sufficiently optimized in terms of reconstruction quality and compression rate.

According to example implementations of the present disclosure, there is provided a context-based

coding solution. Different from generating a predicted image of the target image and performing encoding on the residual between the target image and the predicted image as required in the conventional solution, conditional coding is performed on the target image by extracting contextual feature representation from the reference image in the example implementations of the present disclosure. In this solution, in the feature domain, the contextual information is taken as a condition to guide adaptive encoding on the target image. Such a solution can obtain a higher compression rate in the case of the same bit rate. In addition, since the contextual information in various aspects related to the current image can be characterized in a higher dimension in the feature domain, the context-based image coding can achieve higher reconstruction quality. In this way, performance improvements are achieved in terms of reconstruction quality and compression efficiency.

Hereinafter, some example implementations of the present disclosure will be described in more detail with reference to the accompanying drawings.

Reference is first made to Fig. 2, which illustrates a schematic block diagram of a context-based video coding system 200 according to some implementations of the present disclosure. The system 200 includes an encoder 210, a decoder 220 and a context generator 230.

The encoder 210 is configured to generate an encoded representation y_t , also referred to as a latent code, of an image x_t 202 to be encoded (referred to as a target image in this document). The target image x_t 202 may include a frame in the video segment at time t . In some implementations, the system 200 may further include an entropy model 250, which is configured to perform entropy encoding (on an encoding side) or entropy decoding (on a decoding side). On the encoding side, the entropy model 250 quantizes the encoded representation y_t to obtain a quantized encoded representation $\hat{y}_t$, and determines a bitstream 214 of the target image 202 from the quantized encoded representation $\hat{y}_t$.

On the decoding side, the bitstream 214 corresponding to the target image 202 may be received, and the quantized encoded representation $\hat{y}_t$ may be generated from the bitstream 214. The decoder 220 is configured to generate a decoded image $\hat{x}_t$ 222 corresponding to the target image x_t 202. The decoder 220 may decode the quantized encoded representation $\hat{y}_t$ to determine the decoded image 222.

In some implementations, the encoder 210 and the decoder 220 may be located in the same or different devices, respectively. When they are located in different devices, the different devices

may all include the context generator 230, and may further include the entropy model 250.

According to an example implementation of the present disclosure, the coding of the target image

x_t 202 is based on the reference image 240 thereof. The reference image 240 may include a

decoded image $\hat{x}_{t-1}$ at time t-1 before t in the video segment. On the decoding side, the

5 decoded image $\hat{x}_{t-1}$ may be directly obtained as the reference image 240. On the encoding

side, the decoded image $\hat{x}_{t-1}$ generated by performing the corresponding operation on the

decoding side may be taken as the reference image 240. In other implementations, other images

that are considered to have temporal correlation with the target image x_t 202 may be selected

as the reference images 240. For example, the decoded images at one or more other time before

10 or after t may be selected as the reference images.

The context generator 230 is configured to extract the contextual feature representation 232

(represented as $\bar{x}_t$) of the reference image $\hat{x}_{t-1}$ 240. Assuming that the reference image

$\hat{x}_{t-1}$ 240 and the target image x_t 202 have temporal correlation, the contextual feature

representation $\bar{x}_t$ 232 may characterize the contextual information associated with the target

15 image x_t 202 in the feature domain.

In this document, “feature representation” is characterizing the corresponding feature information

(contextual information here) in the form of a vector, and the vector may have a plurality of

dimensions. “Feature representation” sometimes may be referred to as “vectorized

representation”, “feature vector”, “feature” and so on. These terms are used interchangeably in

20 this document.

In some implementations, the context generator 230 may use a machine learning model to extract

the contextual feature representation $\bar{x}_t$ 232. Some example implementations of contextual

feature extraction will be discussed in more detail with reference to Fig. 4 below.

In the encoding process, the contextual feature representation $\bar{x}_t$ 232 is provided to the encoder

25 210. The encoder 210 is configured to encode the target image x_t 202 based on the contextual

feature representation $\bar{x}_t$ 232. The contextual feature representation $\bar{x}_t$ 232 is provided as a

condition of the encoded target image x_t 202 to help encode better. The encoder 210 is

configured to perform encoding on the target image x_t 202 under the condition of a given

contextual feature representation $\bar{x}_t$ 232, to obtain an encoded representation y_t . Such encoding is also referred to as conditional encoding, and the encoder 210 may be a context encoder. In this document, conditional coding means giving arbitrary information as a condition to help the encoding and decoding of the image.

- 5 Correspondingly, the contextual feature representation $\bar{x}_t$ 232 is provided to the decoder 220 during the decoding process. The decoder 220 is configured to obtain the decoded image 222 corresponding to the target image x_t 202 by decoding based on the contextual feature representation $\bar{x}_t$ 232. The decoder 220 is configured to perform conditional decoding of the target image x_t 202 given the contextual feature representation $\bar{x}_t$ 232. The decoding side also
10 includes a context generator 230. In some implementations, the bitstream 214 is received on the decoding side, and the decoded image 222 is decoded from the bitstream 214 based on the contextual feature representation $\bar{x}_t$ 232.

- Starting from the conventional residual-based coding solution, when it is expected to obtain certain conditions to guide the coding, a direct way might be taking the predicted image $\tilde{x}_t$ of
15 the current target image x_t as the condition. Such conditional coding may be represented as:

$$\hat{x}_t = f_{dec}(\lfloor f_{enc}(x_t | \tilde{x}_t) \rfloor | \tilde{x}_t), \quad (2)$$

where $\tilde{x}_t = f_{predict}(\hat{x}_{t-1})$

- In the above Equation (2), $f_{enc}(x_t | \tilde{x}_t)$ represents the encoding of the target image x_t under the condition of a given predicted image $\tilde{x}_t$, and $f_{dec}(\lfloor f_{enc}(x_t | \tilde{x}_t) \rfloor | \tilde{x}_t)$ represents the decoding of the encoded result under the condition of the given predicted image $\tilde{x}_t$. However, such a condition is still limited by a pixel domain of the image, where each pixel can
20 only be characterized by limited channel dimensions (for example, values of three dimensions RGB). Such a condition will limit the characterization of contextual information.

- In the implementation of the present disclosure, the richer and more relevant contextual information for encoding the target image is characterized by using a higher-dimensional contextual feature representation in the feature domain from the reference image $\hat{x}_{t-1}$ 240. In
25 addition, because the feature representation has a capability of characterizing higher-dimensional

information, different channels in the contextual feature representation 232 may extract different types of contextual information with a higher degree of freedom, including color information, texture information, high-frequency component information, object edge information and so on.

In some implementations, the context-based image coding may be represented as follows:

$$\begin{aligned} \hat{x}_t &= f_{dec}(\lfloor f_{enc}(x_t | \bar{x}_t) \rfloor | \bar{x}_t) \\ \text{where } \bar{x}_t &= f_{context}(\hat{x}_{t-1}) \end{aligned} \quad (3)$$

In the above Equation (3), $f_{enc}(\cdot)$ represents an encoding process of the encoder 210, $f_{dec}(\cdot)$ represents a decoding process of the decoder 220, $f_{context}(\cdot)$ represents a processing operation of the context generator 230, and $\lfloor \cdot \rfloor$ represents the quantization achieved by a rounding operation.

According to an example implementation of the present disclosure, there is provided a context-based image coding solution, especially machine learning-based contextual image coding. In the feature domain, the richer and more relevant contextual information for encoding the target image is characterized by using a higher-dimensional contextual feature representation. Through various contextual features extracted from the contextual feature representation, the context-based image coding can achieve higher reconstruction quality, especially for images with more high-frequency content and complex textures.

Fig. 3 illustrates a capability of the contextual feature representation 232 extracted by the context generator 230 characterizing the contextual information. As shown in Fig. 3, a target image 310 and its reference image 312 are provided. A feature map 320 includes feature maps 321, 322, 323 and 324 of four different channels in the contextual feature representation extracted from the target image 310. These four channels have different emphasis.

The feature map 321 focuses on extracting motion information, because a basketball player in motion shown therein has a higher intensity, and corresponds to a high intensity region in a visual

representation 314 of a motion vector $\hat{m}_t$ between the target image 310 and the reference image 312. As compared with the visual representation 330 with the high-frequency content in the target image 310, it may be seen that the feature map 323 places more emphasis on the high-frequency content to characterize the feature information related to the high-frequency content. In contrast, the feature maps 322 and 324 focus more on color information, where the feature map 322 focuses on green, and the feature map 324 focuses more on red.

The reconstruction error reduction graph 340 in Fig. 3 represents an amount of reduction in reconstruction error that can be obtained by the context-based coding solution according to

example implementations of the present disclosure as compared with the conventional residual-based coding solution. It can be seen from the reconstruction error reduction graph 340 that the context-based coding solution according to example implementations of the present disclosure can achieve significant error reduction, especially the error reduction in the high-frequency regions in the foreground and background. For many conventional codecs, such high-frequency regions are all considered difficult to compress.

In some implementations, the encoder 210 may be configured to perform conditional encoding using an encoding model. The contextual feature representation $\bar{x}_t$ 232 and the target image x_t 202 are provided as an input to the encoding model so that the encoding model processes and outputs the encoded representation corresponding to the target image 202.

In some implementations, the decoder 220 may also be configured to perform conditional decoding using a decoding model. The contextual feature representation $\bar{x}_t$ 232 and the encoded representation corresponding to the target image x_t 202, such as the quantization encoded representation $\hat{y}_t$, are provided as the input to the decoding model so that the decoding model processes and outputs the decoded image 222 corresponding to the target image 202.

The encoding model and the decoding model may be implemented based on various machine learning or deep learning techniques. For example, the encoding model and the decoding model may be based on a neural network (NN), where each model has a plurality of network layers. These network layers for example may include one or more convolutional layers, general normalization (GDN) layers (for encoding model), inverse GDN (IGND) (for decoding model), Resblock layers, etc. In the implementations of the present disclosure, the configuration of the encoding model and the decoding model is not limited.

By using machine learning technology, the encoding model may automatically learn the correlation between the target image x_t 202 and the contextual feature representation $\bar{x}_t$ 232, and reduce the encoding of redundant information based on such correlation, instead of removing the redundancy by a fixed subtraction operation as in the conventional residual-based coding solution.

On the other hand, the encoding model may further adaptively learn how to use the contextual feature representation $\bar{x}_t$ 232. For example, due to the presence of the motion in the video, new content might always appear in an edge region of an object. In this case, since the residual-based coding solution always requires the residual to be encoded, for newly appearing content, the residual is very large and the inter-frame encoding performed via the subtraction operation might

not be as efficient as the intra-frame encoding. On the contrary, the context-based coding according to implementations of the present disclosure can adaptively use the contextual feature representation as a condition. For newly appearing content, the encoding model may adaptively learn to perform intra-frame encoding, thereby significantly improving the compression efficiency. As shown in the reconstruction error reduction graph 340 in Fig. 3, the reconstruction error of the new content appearing in the target image 310 is significantly reduced. It can be seen from the above that the context-based coding according to implementations of the present disclosure can also encode new content caused by the motion very well, and can significantly reduce the reconstruction error.

- 10 In addition to being used for performing encoding and decoding on the target image 202 in the encoder 210 and the decoder 220, in some implementations, the contextual feature representation $\bar{x}_t$ 232 may further be used in the entropy model 250 to perform entropy encoding from the encoded representation generated by the target image 202 to obtain the bitstream 214, or to perform entropy decoding on the bitstream 214 to generate a corresponding quantized encoded representation for decoding by the decoder 220. Example processing regarding the entropy model 250 will be discussed hereunder in more detail with reference to Fig. 5.

Extraction of the contextual feature representation

- In some implementations, the machine learning model used by the context generator 230 may take the reference image $\hat{x}_{t-1}$ 240 as input, and extract the contextual feature representation $\bar{x}_t$ 232 from the reference image $\hat{x}_{t-1}$ 240.
- In some implementations, considering that a video segment often contains various types of content, and might contain many complex motions, motion-related information may also be used to help extract better contextual feature representations $\bar{x}_t$ 232. For example, for a position in the target image x_t 202, the same position in the reference image $\hat{x}_{t-1}$ 240 might have less correlation. In this case, the same position in the feature map of the contextual feature representation $\bar{x}_t$ 232 also has little correlation with that position in the target image x_t 202, and the contextual information with less correlation possibly cannot promote the compression and encoding of the target image x_t 202. Based on this, in some implementations, it is proposed to use motion-related information, such as motion vector (MV) information, to extract contextual feature representation $\bar{x}_t$ 232.

Fig. 4 illustrates a block diagram of an example structure of a context generator 230 in the system

of Fig. 2 according to some implementations of the present disclosure. In the example implementation of Fig. 4, the context generator 230 includes a feature extractor 410 configured to

extract an initial contextual feature representation $\hat{x}_{t-1}$ from the reference image $\hat{x}_{t-1}$ 240.

The feature extractor 410 may be implemented by a machine learning model to convert the reference image 240 from a pixel domain to a feature domain.

The context generator 230 further includes a component for determining motion vector information between the reference image $\hat{x}_{t-1}$ 240 and the target image x_t 202. Fig. 4 shows that the context generator 230 includes a motion estimator 420, an MV encoder 430 and an MV decoder 440 to implement estimation of motion vector information.

The motion estimator 420 is configured to generate motion vector information m_t between time t-1 and time t based on the target image x_t 202. In some examples, the motion estimator 420 may use an optical flow estimation model to determine an optical flow between time t-1 and time t as the motion vector information m_t . The optical flow refers to an instantaneous velocity of a pixel motion of a moving object in a space on an observation imaging plane. Therefore, after the optical flow estimation model is trained, changes of pixels in an image sequence in a time domain and the correlation between adjacent images can be used to find a correspondence relationship between a previous time and the current time, thereby calculating the motion information of the object between the adjacent images. Any currently-existing or future-developed motion vector estimation technique may be used to determine the motion vector information m_t . The implementation of the present disclosure is not limited in this aspect.

The MV encoder 430 is configured to encode the motion vector information m_t to obtain an encoded representation 432 (represented as g_t) of the motion vector information. Similar to the processing of the encoded representation of the target image x_t 202, the encoded representation 432 may be entropy-encoded by the entropy model to obtain the bitstream 430. The bitstream corresponding to the motion vector information may be transmitted to the decoding terminal together with the bitstream of the target image x_t 202. Therefore, on the decoding side, the motion estimator 420 and the MV encoder 430 do not exist. The MV decoder 440 is configured to generate a quantized encoded representation $\hat{g}_t$ for the bitstream 430 of motion vector information m_t , and decode the quantized encoded representation $\hat{g}_t$ to obtain decoded motion

vector information $\hat{m}_t$. The MV encoder 430 and the MV decoder 440 may also be implemented based on a machine learning model.

The context generator 230 further includes a sampling module 450 configured to adjust the initial contextual feature representation $\tilde{x}_{t-1}$ extracted by the feature extractor 410 based on the decoded motion vector information $\hat{m}_t$, so as to extract contextual information more relevant to the target image 202. In some implementations, the sampling module 450 is configured to convert the initial contextual feature representation $\tilde{x}_{t-1}$ through a warping operation to obtain an intermediate contextual feature representation $\ddot{x}_t$. The processing of the sampling module 450 may be represented as $\ddot{x}_t = \text{warp}(\tilde{x}_{t-1}, \hat{m}_t)$, where $\text{warp}()$ represents the warping operation performed by the sampling module 450. The decoded motion vector information $\hat{m}_t$ may be used to guide interpolation sampling for respective element values in the initial contextual feature representation $\tilde{x}_{t-1}$.

The intermediate contextual feature representation $\ddot{x}_t$ may be considered as being capable of characterizing contextual information relatively roughly, because warping operations may introduce some spatial discontinuities. The context generator 230 may also include a context fine-tuning module 460, which is configured to generate a final contextual feature representation $\bar{x}_t$ from the intermediate contextual feature representation $\ddot{x}_t$, where $\bar{x}_t = f_{cr}(\ddot{x}_t)$. The context fine-tuning module 460 may also use a machine learning model to implement fine-tuning of the feature representation. The machine learning model for example may include a plurality of network layers, such as one or more convolutional layers, Resblock layers, and so on. In some implementations, the context-based image coding may be represented as follows:

$$f_{context}(\hat{x}_{t-1}) = f_{cr}(\text{warp}(f_{fe}(\hat{x}_{t-1}), \hat{m}_t)) \quad (4)$$

In the above Equation (4), $f_{fe}()$ represents the feature extraction process of the feature extractor 410, $\text{warp}()$ represents the warping operation performed by the sampling module 450, and $f_{cr}(\cdot)$ represents the context fine-tuning module 460.

The example implementations of extracting the contextual feature representation 232 based on the motion vector information is described above with reference to Fig. 4. It should be appreciated that other methods may also be employed, for example, various other types of machine learning

models may be configured to extract the contextual feature representation from the reference image to facilitate the coding of the target image. The implementation of the present disclosure is not limited in this respect.

Example implementation of the entropy model

- 5 As mentioned briefly above, in some implementations, the contextual feature representation $\bar{x}_t$ 232 may further be used in the entropy model 250 to perform entropy encoding or entropy decoding on the slave image 202. The entropy model is a quantized encoding model commonly used in image coding. On the encoding side, the entropy model 250 can generate the bitstream 214 from the encoded representation y_t output by the encoder 210. On the decoding side, the
- 10 entropy model 250 can determine the quantized encoded representation $\hat{y}_t$ of the target image 202 from the bitstream 214 for further decoding by the decoder 220.

The entropy model mainly considers a cross entropy between an estimated probability distribution and a distribution of the quantized encoded representation $\hat{y}_t$, which is a lower limit value of the actual code rate. This may be represented as:

$$15 \quad R(\hat{y}_t) \geq \mathbb{E}_{\hat{y}_t \sim q_{\hat{y}_t}} [-\log_2 p_{\hat{y}_t}(\hat{y}_t)] \quad (5)$$

where $p_{\hat{y}_t}(\hat{y}_t)$ and $q_{\hat{y}_t}(\hat{y}_t)$ represent the estimated probability quality distribution and the actual probability quality function of the quantized encoded representation $\hat{y}_t$, respectively; $R(\hat{y}_t)$ represents the actual code rate, and $\mathbb{E}_{\hat{y}_t \sim q_{\hat{y}_t}}$ represents the cross entropy.

- In fact, the arithmetic coding can almost encode the quantized encoded representation $\hat{y}_t$ at the code rate of the cross entropy. However, there is still a difference between the actual code rate $R(\hat{y}_t)$ and the cross entropy. Therefore, in some implementations of the present disclosure, the contextual feature representation $\bar{x}_t$ 232 is introduced to enable the entropy model 250 to more accurately estimate the probability distribution $p_{\hat{y}_t}(\hat{y}_t)$ of the latent code.
- 20 Fig. 5 illustrates a block diagram of an example structure of an entropy model 250 in the system of Fig. 2 according to some implementations of the present disclosure. In Fig. 5, the entropy model 250 includes a temporal correlation portion 510, which is configured to determine temporal correlation information between the target image x_t 202 and the reference image $\hat{x}_{t-1}$ 240

based on the contextual feature representation $\bar{x}_t$ 232. The temporal correlation portion 510 may use a temporal priori encoding model 512 to determine the temporal correlation information from the contextual feature representation $\bar{x}_t$ 232. The temporal correlation information can provide temporal prior information, so that the temporal correlation between the processed latent codes can be taken into account.

In addition to the temporal correlation portion 510, the entropy model 250 includes: a typical side information extraction portion 520 for extracting side information from the encoded representation y_t , and a spatial correlation portion 530 for extracting spatial correlation information from the encoded representation y_t . The side information can provide hierarchical prior information in the target image 202, and the spatial correlation information can provide spatial prior information. The side information extraction portion 520 and the spatial correlation portion 530 may be implemented by using modules for extracting the two types of information in the conventional entropy model. Fig. 5 only shows an example implementation regarding the two portions.

As shown in Fig. 5, the side information extraction portion 520 includes: a hyper prior encoder (HPE) 521 for encoding an encoded representation y_t to obtain an intermediate encoded representation z_t ; a quantization (Q) 522 for quantizing the intermediate encoded representation z_t to obtain a quantized encoded representation $\hat{z}_t$; an arithmetic encoder (AE) 523 for quantizing the quantized encoded representation to obtain a bitstream 524 corresponding to the side information; an arithmetic decoder (AD) 525 for decoding the bitstream 524 corresponding to the side information to obtain the quantized encoded representation $\hat{z}_t$; and a hyper prior decoder (HPD) 526 for decoding the arithmetic decoded quantized encoded representation $\hat{z}_t$ to obtain the side information. The bitstream 524 corresponding to the side information may be transmitted to the decoding side.

The entropy model 250 further includes a quantization (Q) 550 for quantizing the encoded representation y_t to obtain a quantized encoded representation $\hat{y}_t$. The quantized encoded representation $\hat{y}_t$ output by the quantization 550 is provided to the spatial correlation portion 530. The spatial correlation portion 530 may use an auto-regressive model 532 to perform acquisition of the spatial correlation information of the target image 202 from the quantized encoded

representation $\hat{y}_t$.

In some implementations, the temporal correlation information, the side information and the spatial correlation information are provided to a priori fusion module 560. The priori fusion module 560 is configured to fuse the temporal correlation information, the side information and

5 the spatial correlation information to determine a mean value μ_t and a variance σ_t of the probability distribution at time t. The mean value μ_t and the variance σ_t may be provided to AE 552. The AE 552 is configured to perform arithmetic encoding on the quantized encoded

representation $\hat{y}_t$ output by the quantization 550 based on the mean value μ_t and the variance σ_t , so as to obtain a bitstream 554 corresponding to the target image 202. The arithmetic encoded

10 representation 554 is provided to the AD 556 which is configured to decode the quantized encoded

representation $\hat{y}_t$ from the bitstream 554 based on the mean value μ_t and the variance σ_t .

In some implementations, the HPE 521, the quantization 522 and AE 523 in the side information extraction portion 520, and the quantization 550 and AE 552 are only included on the encoding side, and may not be needed on the decoding side. The bitstream 524 of the side information

15 extracted by the side information extraction portion 520 may be transmitted to the decoding side for use upon decoding. During decoding, the quantized encoded representation may be determined based on the bitstream 554 corresponding to the target image 202 through the AD 556. In this process, the priori fusion module 560 still provides information on mean value μ_t and the variance

20 σ_t . The quantized encoded representation $\hat{y}_t$ is provided to the encoder 220 to generate a decoded image.

In some implementations, through the processing of the entropy model 250, the determination of $p_{\hat{y}_t}(\hat{y}_t)$ may be represented as follows:

$$p_{\hat{y}_t}(\hat{y}_t|\bar{x}_t, \hat{z}_t) = \prod (\mathcal{L}(\mu_{t,i}, \sigma_{t,i}^2) * \mathcal{U}(-\frac{1}{2}, \frac{1}{2}))(\hat{y}_{t,i}) \quad (6)$$

where $\mu_{t,i}, \sigma_{t,i} = f_{pf}(f_{hpd}(\hat{z}_t), f_{ar}(\hat{y}_{t,<i}), f_{tpe}(\bar{x}_t))$

25 In the above Equation (6), the index i represents a spatial position in the image, assuming that $p_{\hat{y}_t}(\hat{y}_t)$ follows the Laplace distribution. Certainly, it may also be assumed that $p_{\hat{y}_t}(\hat{y}_t)$ follows another distribution, such as Gaussian distribution, mixed Gaussian distribution, and so on. In the above Equation (6), $f_{hpd}(\cdot)$ represents the processing of the HPD 526; $f_{ar}(\cdot)$

represents the processing of the autoregressive model 532; $f_{tpe}^*(\cdot)$ represents the processing of the temporal priori encoding model 512, and $f_{pf}(\cdot)$ represents the processing of the priori fusion module 560.

- It should be appreciated that what is given in Fig. 5 is an example of determining the side information and the temporal correlation information. In other examples, other techniques may also be used to determine the side information and the temporal correlation information. Alternatively or additionally, other information may also be determined, and used together with the temporal correlation information given by the contextual feature representation to perform entropy encoding or entropy decoding on the encoded representation.
- Generally, the extraction of spatial correlation often takes a relatively long period of time. In some implementations, the spatial correlation portion 530 may also be omitted from the entropy model 250. For example, the spatial correlation portion 530 may be bypassed by a switch module 534. The priori fusion module 560 and subsequent modules generate the bitstream 214 based on the temporal correlation information and the side information. The Inventor discovers through many experiments that the omission of spatial correlation information has a very small impact on the reconstruction quality, but may bring about a large improvement in the processing efficiency.

Example implementation of model training

- In the above depictions, many components in the system 200 may be implemented by machine learning models, so it is necessary to determine parameters of these machine learning models through a training process. Various appropriate model training techniques may be employed to implement the training of the machine learning models in the system 200. In some implementations, a trained loss function may be configured based on a distortion of the decoded image and a bit rate overhead. For example, the loss function may be determined as follows:

$$L = \lambda \cdot D + R \quad (7)$$

- where the parameter λ may be a predetermined value for controlling a trade-off between the distortion D and the bit rate overhead R . In some examples, depending on different application requirements, the distortion degree D may be represented with a mean squared error (MSE) or a multi-scale structural similarity (MS-SSIM). In the training process, R may be determined as the cross entropy between the true probability distribution and the estimated probability distribution of the quantized encoded representation.

Example performance comparison

Fig. 6 illustrates the comparison between the context-based coding solution (represented as deep

contextual video compression (DCVC)) according to the present disclosure and four conventional coding solutions from perspective of performance indices in two aspects: the reconstruction quality (represented by PSNR, where PSNR refers to a peak signal-to-noise ratio) and the bit rate overhead BPP (bits per pixel). The four conventional coding solutions are represented as DVC (deep video compression), DVCPPro, x264 and x265 (with the “very slow” configuration level selected), respectively.

Graphs 610, 620, 630, 640, 650 and 660 show measurement of performance indices of five solutions on two video datasets MCL-JCV, UVG, HEVC ClassB, HEVC ClassC, HEVC ClassD and HEVC ClassE, respectively. It can be seen from these graphs that under the same BPP, the context-based coding solution DCVC according to the present disclosure can achieve higher reconstruction quality, namely, PSNR. Under the same PSNR, the context-based coding solution DCVC according to the present disclosure can achieve a lower BPP.

Example processes

Fig. 7 illustrates a flowchart of an image coding process 700 according to some implementations of the present disclosure. The process 700 may be implemented at the system 200 shown in Fig. 2.

At block 700, a reference image of a target image is obtained. At block 720, a contextual feature representation is extracted from the reference image. The contextual feature representation characterizes contextual information associated with the target image. At block 730, conditional encoding or conditional decoding is performed on the target image based on the contextual feature representation.

In some implementations, performing the conditional encoding on the target image comprises: generating an encoded representation of the target image by applying the contextual feature representation and the target image as an input to an encoding model, the encoding model being configured to perform the conditional encoding. In some implementations, decoding the target image comprises: generating a decoded image corresponding to the target image by applying the contextual feature representation and an encoded representation of the target image as an input to a decoding model, the decoding model being configured to perform the conditional decoding.

In some implementations, extracting the contextual feature representation from the reference image comprises: extracting an initial contextual feature representation from the reference image; determining motion vector information between the reference image and the target image; and adjusting the initial contextual feature representation based on the motion vector information, to obtain the contextual feature representation.

In some implementations, performing the conditional encoding or conditional decoding on the target image further comprises: determining temporal correlation information between the target

image and the reference image based on the contextual feature representation; and performing entropy encoding or entropy decoding on the target image at least based on the temporal correlation information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining side information of the target image; and performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the side information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining spatial correlation information of the target image from an encoded representation; and performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the spatial correlation information.

In some implementations, performing the entropy encoding comprises: obtaining an encoded representation of the target image, and generating a bitstream of the target image from an encoded representation of the target image at least based on the temporal correlation information. In some implementations, performing the entropy decoding comprises: obtaining a bitstream of the target image, determining an encoded representation of the target image from the bitstream at least based on the temporal correlation information, and determining a decoded image from the encoded representation of the target image.

Example devices

Fig. 8 illustrates a block diagram of a computing device 800 in which various implementations of the present disclosure may be implemented. It would be appreciated that the computing device 800 as shown in Fig. 8 is merely provided as an example, without suggesting any limitation to the functionalities and scope of implementations of the present disclosure. The computing device 800 may be used to implement an image encoding and/or image decoding process according to implementations of the present disclosure.

As shown in Fig. 8, the computing device 800 includes a computing device 800 in form of a general-purpose computing device. Components of the computing device 800 may include, but are not limited to, one or more processors or processing units 810, a memory 820, a storage device 830, one or more communication units 840, one or more input devices 850, and one or more output devices 860.

In some implementations, the computing device 800 may be implemented as any user terminal or server terminal with computing capability. The server terminal may be any server, large-scale computing device, and the like provided by a variety of service providers. The user terminal may, for example, be any type of mobile terminal, fixed terminal, or portable terminal, including a mobile phone, station, unit, device, multimedia computer, multimedia tablet, Internet node,

communicator, desktop computer, laptop computer, notebook computer, netbook computer, tablet computer, personal communication system (PCS) device, personal navigation device, personal digital assistant (PDA), audio/video player, digital camera/video camera, positioning device, TV receiver, radio broadcast receiver, E-book device, gaming device, or any combination thereof, including the accessories and peripherals of these devices, or any combination thereof. It is also anticipated that the computing device 800 can support any type of interface to a user (such as “wearable” circuitry and the like).

The processing unit 810 may be a physical or virtual processor and may execute various processes based on the programs stored in the memory 820. In a multi-processor system, a plurality of processing units execute computer-executable instructions in parallel so as to enhance parallel processing capability of the computing device 800. The processing unit 810 may also be referred to as a central processing unit (CPU), a microprocessor, a controller, or a microcontroller.

The computing device 800 usually includes various computer storage medium. Such a medium may be any available medium accessible by the computing device 800, including but not limited to, volatile and non-volatile medium, or detachable and non-detachable medium. The memory 820 may be a volatile memory (for example, a register, cache, Random Access Memory (RAM)), non-volatile memory (for example, a Read-Only Memory (ROM), Electrically Erasable Programmable Read-Only Memory (EEPROM), flash memory), or any combination thereof. The memory 820 may include a image coding module 822, which is configured to perform the functionalities of various implementations described herein. The image coding module 822 may be accessed and run by the processing unit 810 to implement the corresponding functions.

The storage device 830 may be any detachable or non-detachable medium and may include machine-readable medium that may be used for storing information and/or data and is accessible within the computing device 800. The computing device 800 may further include additional detachable/non-detachable, volatile/non-volatile memory medium. Although not shown in Fig. 8, there may be provided a disk drive for reading from or writing into a detachable and non-volatile disk, and an optical disk drive for reading from and writing into a detachable non-volatile optical disc. In such case, each drive may be connected to a bus (not shown) via one or more data medium interfaces.

The communication unit 840 implements communication with another computing device via the communication medium. In addition, the functions of components in the computing device 800 may be implemented by a single computing cluster or a plurality of computing machines that may communicate with each other via communication connections. Therefore, the computing device 800 may operate in a networked environment using a logic connection with one or more other servers, personal computers (PCs), or further general network nodes.

The input device 850 may be one or more of a variety of input devices, such as a mouse, keyboard, tracking ball, voice-input device, and the like. The output device 860 may include one or more of a variety of output devices, such as a display, loudspeaker, printer, and the like. By means of the communication unit 840, the computing device 800 may further communicate with one or more external devices (not shown) such as storage devices and display devices, one or more devices that enable the user to interact with the computing device 800, or any devices (such as a network card, a modem and the like) that enable the computing device 800 to communicate with one or more other computing devices, if required. Such communication may be performed via input/output (I/O) interfaces (not shown).

In some implementations, as an alternative of being integrated on a single device, some or all components of the computing device 800 may also be arranged in the form of cloud computing architecture. In the cloud computing architecture, these components may be provided remotely and work together to implement the functionalities described in the present disclosure. In some implementations, cloud computing provides computing, software, data access and storage service, which will not require end users to be aware of the physical locations or configurations of the systems or hardware provisioning these services. In various implementations, the cloud computing provides the services via a wide area network (such as Internet) using proper protocols. For example, a cloud computing provider provides applications over the wide area network, which may be accessed through a web browser or any other computing components. The software or components of the cloud computing architecture and corresponding data may be stored in a server at a remote position. The computing resources in the cloud computing environment may be aggregated at the location of a remote data center or they may be distributed. Cloud computing infrastructure may provide the services through a shared data center, though they behave as a single access point for the users. Therefore, the cloud computing infrastructure may be utilized to provide the components and functionalities described herein from a service provider at remote locations. Alternatively, they may be provided from a conventional server or may be installed directly or otherwise on a client device.

The computing device 800 may be used to implement the context-based image coding in various implementations of the present disclosure. The computing device 800, for example the memory 820 includes an image coding module 822. Upon implementing the image encoding, the image coding module 822 may be configured to perform the above functions regarding the image encoding. Upon image decoding, the image coding module 822 may be configured to perform the above functions regarding the image decoding.

The computing device 800 may receive an input 870 through the input device 850 or communication unit 840. Upon performing encoding, the input 870 includes a target image to be

encoded. Upon performing decoding, the input 807 includes a bitstream to be decoded. The input 870 is provided to the image coding module 822 to perform an image coding operation. Upon performing encoding, the image coding module 822 generates a bitstream of the target image as an output 800. Upon performing decoding, the image coding module 822 generates a decoded
5 image of the target image as the output 800. In some implementations, the output 800 may be output by the output device 860, or may be transmitted to other devices via the communication unit 840.

Example implementations

Some example implementations of the present disclosure are listed below:

- 10 In an aspect, the present disclosure provides a computer-implemented method. The method comprises: obtaining a reference image of a target image; extracting a contextual feature representation from the reference image, the contextual feature representation characterizing contextual information associated with the target image; and performing conditional encoding or conditional decoding on the target image based on the contextual feature representation.
- 15 In some implementations, performing the conditional encoding on the target image comprises: generating an encoded representation of the target image by applying the contextual feature representation and the target image as an input to an encoding model, the encoding model being configured to perform the conditional encoding. In some implementations, performing the conditional decoding on the target image comprises: generating a decoded image corresponding
20 to the target image by applying the contextual feature representation and an encoded representation of the target image as an input to a decoding model, the decoding model being configured to perform the conditional decoding.

In some implementations, extracting the contextual feature representation from the reference image comprises: extracting an initial contextual feature representation from the reference image;
25 determining motion vector information between the reference image and the target image; and adjusting the initial contextual feature representation based on the motion vector information, to obtain the contextual feature representation.

In some implementations, performing the conditional encoding or conditional decoding on the target image further comprises: determining temporal correlation information between the target
30 image and the reference image based on the contextual feature representation; and performing entropy encoding or entropy decoding on the target image at least based on the temporal correlation information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining side information of the target image; and performing the entropy
35 encoding or entropy decoding on the target image at least based on the temporal correlation

information and the side information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining spatial correlation information of the target image from an encoded representation; and performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the spatial correlation information.

In some implementations, performing the entropy encoding comprises: obtaining an encoded representation of the target image, and generating a bitstream of the target image from an encoded representation of the target image at least based on the temporal correlation information. In some implementations, performing the entropy decoding comprises: obtaining a bitstream of the target image, determining an encoded representation of the target image from the bitstream at least based on the temporal correlation information, and determining a decoded image from the encoded representation of the target image.

In another aspect, the present disclosure provides an electronic device. The electronic device comprises: a processor; and a memory coupled to the processor and having instructions stored thereon, the instructions, when executed by the processor, causing the device to perform actions comprising: obtaining a reference image of a target image; extracting a contextual feature representation from the reference image, the contextual feature representation characterizing contextual information associated with the target image; and performing conditional encoding or conditional decoding on the target image based on the contextual feature representation.

In some implementations, performing the conditional encoding on the target image comprises: generating an encoded representation of the target image by applying the contextual feature representation and the target image as an input to an encoding model, the encoding model being configured to perform the conditional encoding. In some implementations, performing the conditional decoding on the target image comprises: generating a decoded image corresponding to the target image by applying the contextual feature representation and an encoded representation of the target image as an input to a decoding model, the decoding model being configured to perform the conditional decoding.

In some implementations, extracting the contextual feature representation from the reference image comprises: extracting an initial contextual feature representation from the reference image; determining motion vector information between the reference image and the target image; and adjusting the initial contextual feature representation based on the motion vector information, to obtain the contextual feature representation.

In some implementations, performing the conditional encoding or conditional decoding on the target image further comprises: determining temporal correlation information between the target image and the reference image based on the contextual feature representation; and performing

entropy encoding or entropy decoding on the target image at least based on the temporal correlation information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining side information of the target image; and performing the entropy
5 encoding or entropy decoding on the target image at least based on the temporal correlation information and the side information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining spatial correlation information of the target image from an encoded representation; and performing the entropy encoding or entropy decoding on the target image at
10 least based on the temporal correlation information and the spatial correlation information.

In some implementations, performing the entropy encoding comprises: obtaining an encoded representation of the target image, and generating a bitstream of the target image from an encoded representation of the target image at least based on the temporal correlation information. In some implementations, performing the entropy decoding comprises: obtaining a bitstream of the target
15 image, determining an encoded representation of the target image from the bitstream at least based on the temporal correlation information, and determining a decoded image based on the encoded representation of the target image.

In a further aspect, the present disclosure provides a computer program product being tangibly stored in a computer storage medium and comprising computer-executable instructions, the
20 computer-executable instructions, when executed by a device, causing the device to perform actions comprising: obtaining a reference image of a target image; extracting a contextual feature representation from the reference image, the contextual feature representation characterizing contextual information associated with the target image; and performing conditional encoding or conditional decoding on the target image based on the contextual feature representation.

In some implementations, performing the conditional encoding on the target image comprises: generating an encoded representation of the target image by applying the contextual feature representation and the target image as an input to an encoding model, the encoding model being configured to perform the conditional encoding; in some implementations, performing the conditional decoding on the target image comprises: generating a decoded image corresponding
25 to the target image by applying the contextual feature representation and an encoded representation of the target image as an input to a decoding model, the decoding model being configured to perform the conditional decoding.

In some implementations, extracting the contextual feature representation from the reference image comprises: extracting an initial contextual feature representation from the reference image;
35 determining motion vector information between the reference image and the target image; and

adjusting the initial contextual feature representation based on the motion vector information, to obtain the contextual feature representation.

In some implementations, performing the conditional encoding or conditional decoding on the target image further comprises: determining temporal correlation information between the target
5 image and the reference image based on the contextual feature representation; and performing entropy encoding or entropy decoding on the target image at least based on the temporal correlation information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining side information of the target image; and performing the entropy
10 encoding or entropy decoding on the target image at least based on the temporal correlation information and the side information.

In some implementations, performing the entropy encoding or entropy decoding on the target image comprises: obtaining spatial correlation information of the target image from an encoded representation; and performing the entropy encoding or entropy decoding on the target image at
15 least based on the temporal correlation information and the spatial correlation information.

In some implementations, performing the entropy encoding comprises: obtaining an encoded representation of the target image, and generating a bitstream of the target image from an encoded representation of the target image at least based on the temporal correlation information. In some implementations, performing the entropy decoding comprises: obtaining a bitstream of the target
20 image, determining an encoded representation of the target image from the bitstream at least based on the temporal correlation information, and determining a decoded image based on the encoded representation of the target image.

In a further aspect, the present disclosure provides a computer readable medium having computer-executable instructions stored thereon, the computer-executable instructions, when executed by a
25 device, causing the device to perform the method in the above aspect.

The functionalities described herein can be performed, at least in part, by one or more hardware logic components. As an example, and without limitation, illustrative types of hardware logic components that can be used include field-programmable gate arrays (FPGAs), application-specific integrated circuits (ASICs), application-specific standard products (ASSPs), system-on-
30 a-chip systems (SOCs), complex programmable logic devices (CPLDs), and the like.

Program code for carrying out the methods of the present disclosure may be written in any combination of one or more programming languages. The program code may be provided to a processor or controller of a general-purpose computer, special purpose computer, or other programmable data processing apparatus such that the program code, when executed by the
35 processor or controller, causes the functions/operations specified in the flowcharts and/or block

diagrams to be implemented. The program code may be executed entirely or partly on a machine, executed as a stand-alone software package partly on the machine, partly on a remote machine, or entirely on the remote machine or server.

- In the context of the present disclosure, a machine-readable medium may be any tangible medium
- 5 that may contain or store a program for use by or in connection with an instruction execution system, apparatus, or device. The machine-readable medium may be a machine-readable signal medium or a machine-readable storage medium. A machine-readable medium may include but is not limited to an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, or device, or any suitable combination of the foregoing. More specific examples of the
- 10 machine-readable storage medium would include an electrical connection having one or more wires, a portable computer diskette, a hard disk, a random-access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage device, a magnetic storage device, or any suitable combination of the foregoing.
- 15 Further, while operations are depicted in a particular order, this should not be understood as requiring that such operations are performed in the particular order shown or in sequential order, or that all illustrated operations are performed to achieve the desired results. In certain circumstances, multitasking and parallel processing may be advantageous. Likewise, while several specific implementation details are contained in the above discussions, these should not
- 20 be construed as limitations on the scope of the present disclosure, but rather as descriptions of features that may be specific to particular implementations. Certain features that are described in the context of separate implementations may also be implemented in combination in a single implementation. Rather, various features described in a single implementation may also be implemented in multiple implementations separately or in any suitable sub-combination.
- 25 Although the subject matter has been described in language specific to structural features and/or methodological actions, it is to be understood that the subject matter specified in the appended claims is not necessarily limited to the specific features or actions described above. Rather, the specific features and actions described above are disclosed as example forms of implementing the claims.

CLAIMS

1. An image coding method comprising:
obtaining a reference image of a target image;
extracting a contextual feature representation from the reference image, the contextual feature representation characterizing contextual information associated with the target image; and
performing conditional encoding or conditional decoding on the target image based on the contextual feature representation.
2. The method of claim 1,
wherein performing the conditional encoding on the target image comprises:
generating an encoded representation of the target image by applying the contextual feature representation and the target image as an input to an encoding model, the encoding model being configured to perform the conditional encoding; or
wherein performing the conditional decoding on the target image comprises:
generating a decoded image corresponding to the target image by applying the contextual feature representation and an encoded representation of the target image as an input to a decoding model, the decoding model being configured to perform the conditional decoding.
3. The method of claim 1, wherein extracting the contextual feature representation from the reference image comprises:
extracting an initial contextual feature representation from the reference image;
determining motion vector information between the reference image and the target image;
and
adjusting the initial contextual feature representation based on the motion vector information, to obtain the contextual feature representation.
4. The method of claim 1, wherein performing the conditional encoding or conditional decoding on the target image further comprises:
determining temporal correlation information between the target image and the reference image based on the contextual feature representation; and
performing entropy encoding or entropy decoding on the target image at least based on the temporal correlation information.
5. The method of claim 4, wherein performing the entropy encoding or entropy decoding on the target image comprises:
obtaining side information of the target image; and
performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the side information.
6. The method of claim 4, wherein performing the entropy encoding or entropy

decoding on the target image comprises:

obtaining spatial correlation information of the target image from an encoded representation; and

performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the spatial correlation information.

7. The method of claim 4,

wherein performing the entropy encoding comprises:

obtaining an encoded representation of the target image, and

generating a bitstream of the target image from an encoded representation of the target image at least based on the temporal correlation information, and

wherein performing the entropy decoding comprises:

obtaining a bitstream of the target image,

determining an encoded representation of the target image from the bitstream at least based on the temporal correlation information, and

determining a decoded image from the encoded representation of the target image.

8. An electronic device, comprising:

a processor; and

a memory coupled to the processor and having instructions stored thereon, the instructions, when executed by the processor, causing the device to perform actions comprising:

obtaining a reference image of a target image;

extracting a contextual feature representation from the reference image, the contextual feature representation characterizing contextual information associated with the target image; and

performing conditional encoding or conditional decoding on the target image based on the contextual feature representation.

9. The electronic device of claim 8,

wherein performing the conditional encoding on the target image comprises:

generating an encoded representation of the target image by applying the contextual feature representation and the target image as an input to an encoding model, the encoding model being configured to perform the conditional encoding; or

wherein performing the conditional decoding on the target image comprises:

generating a decoded image corresponding to the target image by applying the contextual feature representation and an encoded representation of the target image as an input to a decoding model, the decoding model being configured to perform the conditional decoding.

10. The electronic device of claim 8, wherein extracting the contextual feature

representation from the reference image comprises:

- extracting an initial contextual feature representation from the reference image;
- determining motion vector information between the reference image and the target image;
- and
- adjusting the initial contextual feature representation based on the motion vector information, to obtain the contextual feature representation.

11. The electronic device of claim 8, wherein performing the conditional encoding or conditional decoding on the target image further comprises:

- determining temporal correlation information between the target image and the reference image based on the contextual feature representation; and
- performing entropy encoding or entropy decoding on the target image at least based on the temporal correlation information.

12. The electronic device of claim 11, wherein performing the entropy encoding or entropy decoding on the target image comprises:

- obtaining side information of the target image; and
- performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the side information.

13. The electronic device of claim 11, wherein performing the entropy encoding or entropy decoding on the target image comprises:

- obtaining spatial correlation information of the target image from an encoded representation; and
- performing the entropy encoding or entropy decoding on the target image at least based on the temporal correlation information and the spatial correlation information.

14. The electronic device of claim 11, wherein performing the entropy encoding comprises:

- obtaining an encoded representation of the target image, and
- generating a bitstream of the target image from an encoded representation of the target image at least based on the temporal correlation information, and

wherein performing the entropy decoding comprises:

- obtaining a bitstream of the target image,
- determining an encoded representation of the target image from the bitstream at least based on the temporal correlation information, and
- determining a decoded image based on the encoded representation of the target image.

15. A computer program product being tangibly stored in a computer storage medium

and comprising computer-executable instructions, the computer-executable instructions, when executed by a device, causing the device to perform actions comprising:

obtaining a reference image of a target image;

extracting a contextual feature representation from the reference image, the contextual feature representation characterizing contextual information associated with the target image; and

performing conditional encoding or conditional decoding on the target image based on the contextual feature representation.