

IMAGE PROCESSING METHOD AND APPARATUS, DEVICE, AND STORAGE MEDIUM

TECHNICAL FIELD

[0001] The disclosure relates, but is not limited, to the technical field of computer vision, and particularly to an image processing method and apparatus, a device, and a storage medium.

BACKGROUND

[0002] Portrait video stylization aims to render a portrait region in a video into a specified artistic style, and is applied in daily production and life extensively to avatar design, animation poster advertising production, virtual uploader operation, etc.

[0003] At present, a mainstream technology of the industry uses an image conversion method to train a network to learn a mapping relationship between a real face image and an artistic portrait image. Such a method depends on a large amount of training data, and is only suitable for small images, thus difficult to meet practical requirements, and more inapplicable to the field of videos. Another technology performs transfer learning on a small artistic portrait image data set based on a Style-Based Generator Architecture for Generative Adversarial Networks (StyleGAN) to implement high-definition portrait stylization. However, the StyleGAN is able to process images of a fixed size only, and requires face alignment, so this technology is also inapplicable to the field of videos.

[0004] How to design a framework to support high-definition portrait video stylization without face alignment is a problem.

SUMMARY

[0005] In view of this, embodiments of the disclosure at least provide an image processing method and apparatus, a device, and a storage medium.

[0006] The technical solutions of the embodiments of the disclosure are implemented as follows.

[0007] In a first aspect, the embodiments of the disclosure provide a method for training an image processing model. An image processing model may include a trained style encoder, a first generative subnetwork to be trained, and a trained second generative subnetwork. The style encoder may be configured to map a style exemplar or real sample into a style code. The first generative subnetwork may include a fully convolutional network, and may be configured to generate a first content feature of the real sample. The second generative subnetwork may be configured to generate a style image of the real sample based on the style code and the first content feature. The method may include the following operations. A style sample set and a real sample set are acquired. The first generative subnetwork is trained based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image. After training of the first generative subnetwork completes, the image processing model is trained iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied.

[0008] In a second aspect, the embodiments of the disclosure provide an image processing method, which can be applied to an image processing model. The image processing model may include a style encoder, a first generative subnetwork and a second generative subnetwork. The style encoder may be configured to map a style exemplar or content image to a style code. The first generative subnetwork may include a fully convolutional network, and may be configured to generate a first content feature of the content image. The second generative subnetwork may be configured to generate a style image of the content image based on the style code and the first content feature. The method may include the following operations. The style encoder generates the style code of the content image, the content image being an image determined from an image frame set to be processed, and the image frame set having image frames with a sequential relationship. The first generative subnetwork generates, based on the content image, the

first content feature capable for reconstruction of a structural feature of the style image. The second generative subnetwork generates the style image based on the first content feature and the style code.

[0009] In some embodiments, the operation that the style encoder generates the style code of the content image may include the following operations. Face detection is performed on the content image to obtain a face detection box in the content image. A detection region of a preset size is obtained by cropping taking a center of the face detection box as a center. The style code of the content image is generated by using the style encoder based on the detection region.

[0010] In a third aspect, the embodiments of the disclosure provide an apparatus for training an image processing model. The image processing model may include a style encoder, a first generative subnetwork and a second generative subnetwork. The style encoder may be configured to map a style exemplar or real sample into a style code. The first generative subnetwork may include a fully convolutional network, and may be configured to generate a first content feature of the real sample. The second generative subnetwork may be configured to generate a style image of the real sample based on the style code and the first content feature. The apparatus may include a first acquisition module, a first training module, and a second training module. The first acquisition module may be configured to acquire a style sample set and a real sample set. The first training module may be configured to train the first generative subnetwork based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image. The second training module may be configured to, after training of the first generative subnetwork completes, train the image processing model iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied.

[0011] In a fourth aspect, the embodiments of the disclosure provide an image processing apparatus, which can be applied to an image processing model. The image processing model may include a style encoder, a first generative subnetwork and a second generative subnetwork. The style encoder may be configured to map a style

exemplar or content image to a style code. The first generative subnetwork may include a fully convolutional network, and may be configured to generate a first content feature of the content image. The second generative subnetwork may be configured to generate a style image of the content image based on the style code and the first content feature. The apparatus may include a first generation module, a reconstruction module, and a second generation module. The first generation module may be configured to generate the style code of the content image by using the style encoder, the content image being an image determined from an image frame set to be processed, and the image frame set having image frames with a sequential relationship. The reconstruction module may be configured to generate, based on the content image by using first generative subnetwork, the first content feature capable for reconstruction of a structural feature of the style image. The second generation module may be configured to generate the style image by using second generative subnetwork based on the first content feature and the style code.

[0012] In a fifth aspect, the embodiments of the disclosure provide a computer device, which may include a memory and a processor. The memory may store a computer program capable of running in the processor. The processor may execute the program to implement part or all of the steps in the method.

[0013] In a sixth aspect, the embodiments of the disclosure provide a computer-readable storage medium, which may store a computer program executed by a processor to implement part or all of the steps in the method.

[0014] In the related art, a generator, such as a StyleGAN, is able to process images of a fixed size only, and requires to perform face alignment on the processed images. The image processing model provided in the embodiments of the disclosure includes a trained style encoder, a first generative subnetwork to be trained, and a trained second generative subnetwork. A first content feature generated by the first generative subnetwork is made capable for reconstruction of a structural feature of a style image, whereby first seven layers of sub-networks in the generator, such as the StyleGAN, are replaced with the first generative subnetwork. The first generative subnetwork includes a fully convolutional network. According to the translation invariance of the convolutional network, face alignment is not needed for an image processed by the image processing model. In

addition, according to the reusability of a convolutional kernel in the convolutional network, the image processing model may process images of any size. Therefore, the problem in the related art that only images of a fixed size can be processed and face alignment is required is solved, the image processing model can be applied to the field of videos, the application range of the image processing model is extended, and ease of use of the image processing model is improved.

[0015] It is to be understood that the above general description and the following detailed description are only exemplary and explanatory and not intended to limit the technical solutions of the disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate embodiments consistent with the disclosure and, together with the specification, serve to describe the technical solutions of the disclosure.

[0017] FIG. 1A is a composition structure diagram of a StyleGAN model.

[0018] FIG. 1B is a flowchart of a method for training an image processing model according to an embodiment of the disclosure.

[0019] FIG. 2 is an implementation flowchart of S200 according to an embodiment of the disclosure.

[0020] FIG. 3 is an implementation flowchart of S3001 according to an embodiment of the disclosure.

[0021] FIG. 4A is a composition structure diagram of an image processing model according to an embodiment of the disclosure.

[0022] FIG. 4B is a schematic diagram of overlapping regions between style images of different image frames in a real sample according to an embodiment of the disclosure.

[0023] FIG. 5A is a composition structure diagram of another image processing model

according to an embodiment of the disclosure.

[0024] FIG. 5B is an implementation flowchart of an image processing method according to an embodiment of the disclosure.

[0025] FIG. 6 is a composition structure diagram of an apparatus for training an image processing model according to an embodiment of the disclosure.

[0026] FIG. 7 is a composition structure diagram of an image processing apparatus according to an embodiment of the disclosure.

[0027] FIG. 8 is a schematic diagram of a hardware entity of a computer device according to an embodiment of the disclosure.

DETAILED DESCRIPTION

[0028] In order to make the objectives, technical solutions, and advantages of the disclosure clearer, the technical solutions of the disclosure will further be described below in combination with the drawings and embodiments in detail. The described embodiments should not be considered as limitations on the disclosure. All other embodiments obtained by those of ordinary skill in the art without creative work shall fall within the scope of protection of the disclosure.

[0029] "Some embodiments" involved in the following descriptions describes a subset of all possible embodiments. However, it can be understood that "some embodiments" may be the same or different subsets of all the possible embodiments, and may be combined with one another without conflicts.

[0030] Involved term "first/second/third" is only for distinguishing similar objects, and does not represent a specific sequence of the objects. It can be understood that "first/second/third" may be interchanged to specific sequences or orders if allowed, so as to implement the embodiments of the disclosure described herein in sequences except those illustrated or described herein.

[0031] Unless otherwise defined, all technical and scientific terms used herein have the same meanings as commonly understood by those skilled in the art the disclosure belongs to. Terms used herein are only for a purpose of describing the disclosure and not intended to limit the disclosure.

[0032] Nouns and terms involved in the embodiments of the disclosure will be described first before the embodiments of the disclosure are further described in detail. The nouns and terms involved in the embodiments of the disclosure are suitable to be explained as follows.

[0033] The Adaptive Instance Normalization (AdaIN) layer is a network module. This module is configured to normalize image features according to a channel dimension so as to normalize an average value and variance of the image features in each channel to 0 to 1.

[0034] The perceptual loss is used to evaluate a difference between feature maps of a generated style image and a style sample.

[0035] StyleGAN is a style-based generative network capable of mapping a random style code to a face image. The StyleGAN may convert a generation space of the generative network from a face image domain to an artistic face image domain through hundreds of artistic face images and several hours of training.

[0036] The content image is an image providing a content in an image stylization process. It is generally an image input by a user, such as a real face image. In the subsequent description, capital letter I is used to represent a content image. In different contexts, I may refer to a real face image. For a training process, I may refer to a real sample.

[0037] The style image may be an image of any object obtained by exemplar stylization, such as a face, an animal and a scenery. Since face style images are involved in most cases, for ease of understanding, descriptions are made in the following embodiments of the disclosure taking face stylization as an example. In such case, the style image can be understood as an artistic portrait image. In the subsequent description, capital letter S is used to represent a style image. In different contexts, S may refer to an

artistic portrait image. For a training process, S may refer to a style sample.

[0038] Before the embodiments of the disclosure are introduced, a StyleGAN model is introduced first. As shown in FIG. 1A, the model 10 includes a mapping network 11 and a generative network 12.

[0039] The mapping network 11 may map a style code of an input content image or a randomly sampled style code z to an internal intermediate style code (which can be understood as an intermediate vector), and subsequently transmit the internal intermediate style code to the generative network 12 as 18 control vectors of the generative network 12 such that different elements of the control vector may control different visual features. The style code of the content image may be obtained by coding the content image I by an encoder 111.

[0040] The mapping network 11 includes eight Full Connection (FC) layers. The internal intermediate style code $\hat{W}$ output by the mapping network 11 is the same as an input of the StyleGAN model in size. Generally, the size of the input may be (512×1) or (512×18) . That is, the input is one 512-dimensional vector, or eight 512-dimensional vectors. If the input is one 512-dimensional vector, an input layer may generate 18 duplicates of this vector.

[0041] The generative network 12 includes nine generation modules for transformation of 4×4 to 1024×1024 . That is, each module corresponds to transformation of 4×4 , 8×8 , 16×16 , 32×32 , 64×64 , 128×128 , 256×256 , 512×512 and 1024×1024 . Except the first generation module and the last generation module, each module may be affected by two control vectors (input of affine transformation A in the figure). The module is affected by one of the two control vectors after upsampling and by the other control vector after convolution, both through a style adjustment module (AdaIN layer). The first module performs no upsampling operation but only a convolution operation, so there is only one control vector for the first module. For the last module, there are control vectors corresponding to the upsampling operation and the convolution operation as well as a control vector corresponding to transformation to a three-channel image. Therefore, the internal intermediate style code $\hat{W}$ is transformed into totally 18 control vectors for transmission to the generative network 12.

[0042] From another angle, the first generation module includes an affine transformation module, a convolutional layer, and a style adjustment module, but no upsampling module. Each of the other generation modules includes an upsampling module, an affine transformation module, a convolutional layer, and a style adjustment module.

[0043] As shown in FIG. 1A, the first generation module implements style adjustment through the following process. After passing through the learning affine transformation module A (which may be an FC layer), the internal intermediate style code $\hat{W}$ is expanded and transformed into a scaling factor $y_{(s, i)}$ and a deviation factor $y_{(b, i)}$ through the style adjustment module. The scaling factor is multiplied by a feature output by the convolutional layer after standardization. Then, a product is summed with the deviation factor. As such, a process of affecting a style image by the internal intermediate style code $\hat{W}$ is completed.

[0044] In the StyleGAN, if a resolution of a layer is lower, affected features are rougher. These features may be divided into two types: 1) rough features, and 2) fine features. The rough features have resolutions of not higher than 32×32 , correspond to resolution layers 1 to 7, and affect the posture, the hair style, the facial shape, etc., as well as facial features, the hair style, opening or closing of the eyes, etc. The fine features have resolutions of 64×64 to 1024×1024 , correspond to resolution layers 8 to 18, and affect colors (the eyes, the hair, and the skin) and microscopic features.

[0045] The StyleGAN may achieve relatively high performance in generating artistic portrait images by transfer learning of finite data. However, this method can process images of a fixed size only and requires face alignment, thus is inapplicable to the field of videos.

[0046] For the foregoing problem, the embodiments of the disclosure provide an image processing model. Compared with the StyleGAN, the image processing model provided in the embodiments of the disclosure is obtained by replacing the first layers of subnetworks in the generative network 12 with a first generative subnetwork including a fully convolutional network. According to the translation invariance of the fully convolutional network and the reusability of a convolutional kernel in the fully

convolutional network, the image processing model can process images of any size without face alignment, and thus can be applied to the field of videos.

[0047] The embodiments of the disclosure provide a method for training an image processing model, which may be executed by a processor of a computer device. The computer device may refer to any proper device with a data processing capability, such as a server, a notebook computer, a tablet computer, a desktop computer, a smart television, a set-top box, and a mobile device (such as a mobile phone, a portable video player, a personal digital assistant, a dedicated messaging device, and a portable gaming device).

[0048] An image processing model includes a trained style encoder (i.e., the above-mentioned mapping network 11), a first generative subnetwork to be trained (equivalent to the first seven layers of subnetworks in the above-mentioned generative network 12), and a trained second generative subnetwork (i.e., the latter 11 layers of subnetworks in the above-mentioned generative network 12). The style encoder is configured to map a style exemplar or real sample I into a style code (i.e., the above-mentioned internal intermediate style code). The first generative subnetwork includes a fully convolutional network, and is configured to generate a first content feature of the real sample I. The second generative subnetwork is configured to generate a style image of the real sample I based on the style code and the first content feature. Referring to FIG. 1B, the method at least includes the following operations S100, S200, and S300.

[0049] In S100, a style sample set and a real sample set are acquired.

[0050] Here, the real sample set may be a real image data set, such as an image data set obtained by shooting like a real face data set, a real animal data set, and a real scenery data set.

[0051] The style sample set may be an artistic image data set. Artistic images in the artistic image data set may be artistic portrait images, artistic scenery images, cartoon images, animation images, etc. For example, the images in the artistic image data set may be images of different forms obtained by rendering, morphing, and/or other modification modes.

[0052] In some embodiments, the implementation of S100 may include that: a first

generator G0 pre-trained based on the real sample set is acquired; and the first generator is trained based on the style sample set to obtain a second generator G1. During implementation, the first generator is finely tuned to obtain the second generator G1. G0 may be a whole StyleGAN (including a mapping network and a generative network) rather than the generative network only. Then, the real sample set and style sample set for training are generated by using the models G0 and G1 through a randomly sampled Gaussian noise z . During implementation, the randomly sampled Gaussian noise z is input to the model G0 and the model G1 respectively such that the model G0 generates a real portrait image $x'=G0(z)$ and the model G1 generates a corresponding artistic portrait image $y'=G1(z)$, and the same geometric-transformation-based data enhancement, such as translation, rotation, and scaling, is performed on x' and y' to obtain unaligned data (x, y) , i.e., a real sample and a style sample.

[0053] In S200, the first generative subnetwork is trained based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image.

[0054] Here, the first generative subnetwork is trained based on the real sample set. Features of the style image include structural features (i.e., rough features generated by the first seven layers of the above-mentioned generative network 12) and color features (fine features generated by the latter 11 layers of the above-mentioned generative network 12). In S200, making the first content feature generated by the first generative subnetwork capable for reconstruction of the structural feature of the style image refers to making the first content feature generated by the first generative subnetwork similar to or the same as the rough features generated by the first seven layers of subnetworks of the generative network 12 in the StyleGAN, whereby the style image of the real sample may be generated in combination with the fine features generated by the second generative subnetwork (i.e., the latter 11 layers of subnetworks in the above-mentioned generative network 12).

[0055] In some embodiments, as shown in FIG. 2, the implementation of S200 may include the following operations S2001 to S2005.

[0056] In S2001, the style code is generated by using the style encoder, the style code

including a structural style code and a color style code.

[0057] Here, the structural style code is a code for the structural feature of the style image, namely corresponding to the first seven layers of subnetworks of the above-mentioned generative network 12, and is used to control styles of the structural features generated by the first seven layers of subnetworks of the generative network 12. The color style code is a code for the color feature of the style image, namely corresponding to the latter 11 layers of subnetworks of the generative network 12, and is used to control the color features generated by the latter 11 layers of subnetworks of the generative network 12.

[0058] The style encoder is a trained model whose input may be a style exemplar or a real sample. The style encoder may map the style exemplar or the real sample to a style code. When the input of the style encoder is a style exemplar, it is necessary to convert the style exemplar to a fixed size that the style encoder can process, such as 256*256, and then the style exemplar is input to the style encoder for style encoding.

[0059] When the input of the style encoder is a real sample, the implementation of S2001 may include the following operations S21a to S21c.

[0060] In S21a, face detection is performed on a real sample in the real sample set to obtain a face detection box in the real sample.

[0061] Here, during the implementation of 21a, face detection may be performed on the real sample in the real sample set by using a target detection algorithm, such as RetinaNet, YOLO, Fast RCNN, and Faster RCNN. An output detection result is the face detection box in the real sample.

[0062] In S21b, a detection region of a preset size is obtained by cropping taking a center of the face detection box as a center.

[0063] Here, the style encoder needs to process images of a fixed size. Therefore, during the implementation of 21b, a detection region of the size that the style encoder may process is obtained by cropping taking a center of the face detection box as a center.

[0064] In S21c, the style code is generated by using the style encoder based on the detection region.

[0065] In the embodiment of the disclosure, when the input of the style encoder is a real sample, after a face detection box in the real sample is detected, a detection region of a preset size is obtained by cropping taking a center of the face detection box as a center, and the style code is generated by taking the detection region as the input of the style encoder. As such, generation of the style code based on the real sample is implemented.

[0066] In S2002, a second content feature is generated by using a third generative subnetwork of a generator based on the structural style code and the real sample set, the generator including the style encoder, the third generative subnetwork and the second generative subnetwork, and the third generative subnetwork being configured to affect a structural feature of a style image corresponding to a real sample of a fixed size.

[0067] Here, the generator is the original StyleGAN, and is configured to process a real sample of a fixed size. The third generative subnetwork in the generator is configured to affect a structural feature of a style image corresponding to the real sample of the fixed size. The second content feature is a feature generated by the third generative subnetwork (corresponding to the first seven layers of subnetworks of the generative network 12 in FIG. 1A) in the original StyleGAN.

[0068] In S2003, the first content feature is generated by using the first generative subnetwork based on the real sample set.

[0069] Here, the first content feature is a feature generated by the first generative subnetwork including the fully convolutional network in a case where the input is the real sample set. The first generative subnetwork is of different network structures in case of different manners used by the image processing model for generating the style image. The image processing model generates the style image in two manners: overall-style-based portrait stylization (namely generating one style) and style-exemplar-based portrait stylization (namely generating images of different styles according to style exemplars). Correspondingly, the first generative subnetwork includes two different network structures.

[0070] The first network structure (for overall-style-based portrait stylization): the first generative subnetwork may include a feature extraction module and a residual module. The feature extraction module is configured to extract the real sample to obtain features under different resolutions. The residual module is configured to extract deeper features. Correspondingly, the implementation of the operation in S2003 that “the first content feature is generated by using the first generative subnetwork based on the real sample set” may include the following operations S213a to S213b.

[0071] In S213a, feature extraction is performed on the real sample by using the feature extraction module to obtain third content features under different resolutions.

[0072] Here, during the implementation of 213a, third content features under different resolutions may be obtained as needed. For example, if a size of the real sample in pixels is $W \times H$, the feature extraction module may perform feature extraction on the real sample to obtain third content features under three resolutions of $W/2 \times H/2$, $W/4 \times H/4$, and $W/8 \times H/8$ as needed.

[0073] In S213b, feature extraction is performed on the third content feature under a minimum resolution of the different resolutions by using the residual module to obtain the first content feature.

[0074] Here, the third content feature under the minimum resolution is an input of the residual module, and is used for the residual module to extract a deeper feature. For example, if the minimum resolution is $W/8 \times H/8$, the third content feature under the resolution of $W/8 \times H/8$ in the third content features under the three resolutions of $W/2 \times H/2$, $W/4 \times H/4$, and $W/8 \times H/8$ is the input of the residual module, and the residual module performs further feature extraction on the third content feature under $W/8 \times H/8$ to obtain the first content feature. Here, the residual module is used to increase the network depth without loss of the input feature. Since a deeper network has a higher learning capability and may extract features better, the quality of the generated style image may be improved.

[0075] It is to be noted that the first generative subnetwork is for generating a deep first content feature under a target resolution. When another fully convolutional network

may achieve the same effect, the first generative subnetwork may also be the other fully convolutional network. The network structure of the first generative subnetwork is not limited in the embodiment of the disclosure.

[0076] The second network structure (for style-exemplar-based portrait stylization): the first generative subnetwork may include a feature extraction module, a residual module, and a residual adjustment module. Correspondingly, the implementation of the operation in S2003 that “the first content feature is generated by using the first generative subnetwork based on the real sample set” may include the following operations S223a to S223d.

[0077] In S223a, a style degree of the style exemplar is acquired.

[0078] Here, the style degree is used to represent a variation of the style exemplar relative to a real image. If a value of the style degree is greater, it indicates that the variation of the style exemplar relative to the real image is greater.

[0079] In S223b, feature extraction is performed on the real sample set by using the feature extraction module to obtain third content features under different resolutions.

[0080] Here, for S223b, reference is made to S213a.

[0081] In S223c, feature extraction is performed on the third content feature under a minimum resolution of the different resolutions by using the residual module to obtain a second intermediate feature.

[0082] Here, for S223c, reference is made to S213b.

[0083] In S223d, the second intermediate feature is adjusted by using the residual adjustment module based on the style degree and the style code to generate the first content feature.

[0084] Here, the implementation of S223d may include that: by using the residual adjustment module, the second intermediate feature is summed with the style code, and a sum is multiplied by the style degree to obtain the first content feature. In some embodiments, the residual adjustment module includes six submodules, and each residual

adjustment submodule inputs a style code under a resolution (the style code includes two style codes under each of resolutions of $8*8$, $16*16$, and $32*32$). The residual module also includes six submodules, and each residual submodule generates a second intermediate feature. During implementation, the second intermediate feature generated by the present residual submodule may be summed with the style code input by the corresponding residual adjustment submodule to obtain a sixth feature. Then, the sixth feature is multiplied by the style degree to obtain an output feature of the present residual submodule as an input feature of the next residual submodule for generating the second intermediate feature of the next residual submodule. The operations are cyclically repeated until an output feature of the last residual submodule is generated as the first content feature. In this manner, the generated first content feature is fused with the structural feature of the style exemplar, so that the finally generated style image is of a style of the style exemplar.

[0085] In some embodiments, since the second intermediate feature is inconsistent with the resolution of the style code, a convolutional layer in the residual adjustment module may be replaced with atrous convolution, with a convolutional kernel remaining unchanged, to match a receptive field by changing an expansion rate. For example, an expansion rate of the residual adjustment module corresponding to the style code under the resolution of $8*8$ is set to 4, so that the style code under the resolution of $8*8$ may be summed with the second intermediate feature under the resolution of $32*32$.

[0086] In S2004, a first regularization loss between each first content feature and the corresponding second content feature is determined, the first regularization loss being used to represent a difference between the first content feature and the second content feature.

[0087] Here, the first regularization loss is determined for a purpose of reducing the difference between the first content feature and the second content feature, namely making the first content feature similar to the second content feature as much as possible, such that the first generative subnetwork may replace the third generative subnetwork (i.e., the first seven layers of subnetworks of the generative network 12 in FIG. 1A) in the original StyleGAN. An expression of a first regularization loss function may be:

[0088] $L_E = \|\mathbf{f}'_E(x'_\downarrow) - \mathbf{f}_{G_1}(z)\|.$

[0089] $\mathbf{f}'_E(x'_\downarrow)$ represents a feature generated by the first generative subnetwork, i.e., the first content feature. $\mathbf{f}_{G_1}(z)$ represents a feature generated by the third generative subnetwork, i.e., the second content feature. $x'_\downarrow$ represents an input image of the first generative subnetwork. z represents an input vector of the third generative subnetwork.

[0090] In S2005, the first generative subnetwork is trained iteratively based on the first regularization loss until a second convergence condition is satisfied.

[0091] Here, the first content feature generated by the trained first generative subnetwork is very close to the second content feature generated by the third generative subnetwork in the original StyleGAN, and thus may replace the second content feature generated by the third generative subnetwork in the original StyleGAN.

[0092] In the embodiment of the disclosure, the second content feature is generated first by using the third generative subnetwork of the generator based on the structural style code and the real sample set. Then, the first content feature is generated by using the first generative subnetwork based on the real sample set. Next, the first regularization loss between each first content feature and the corresponding second content feature is determined to train the first generative subnetwork iteratively until the second convergence condition is satisfied. In this manner, the first content feature generated by the first generative subnetwork is very close to the second content feature generated by the third generative subnetwork in the original StyleGAN, and may replace the second content feature generated by the third generative subnetwork in the original StyleGAN. Since the second content feature is used to affect the structural feature of the style image, the first content feature may reconstruct the structural feature of the style image.

[0093] In S300, after training of the first generative subnetwork completes, the image processing model is trained iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied.

[0094] In some embodiments, the implementation of S300 may include the following operations S3001 to S3003.

[0095] In S3001, the image processing model is trained iteratively based on the style sample set, the real sample set, and the style code to obtain each style image generated by the image processing model.

[0096] Here, the style code includes a structural style code and a color style code. In some embodiments, the implementation of S3001 may include the following two conditions according to the manner the image processing model generates the style image.

[0097] Under the first condition (overall-style-based portrait stylization), style encoding is performed on the real sample to generate a color style code. Then, the real sample set is input to the first generative subnetwork to generate the first content feature. Finally, each style image is generated by using the second generative subnetwork based on the first content feature and the color style code. Under this condition, the generated style image is of the same style.

[0098] Under the second condition (style-exemplar-based portrait stylization), style encoding is performed on the style exemplar to generate a structural style code and a color style code. Then, the real sample set is input to the first generative subnetwork to generate an intermediate feature. Then, the intermediate feature is adjusted by using the structural style code and the style degree of the style exemplar to generate the first content feature. Finally, each style image is generated by using the second generative subnetwork based on the first content feature and the color style code. Under this condition, the generated style image may be adjusted according to the style exemplar. That is, the generated style image may include multiple styles.

[0099] Under the two conditions, the style sample set is used as a truth value of each generated style image to train the image processing model until the image processing model meets a convergence requirement.

[00100] In S3002, a total loss of the image processing model is determined based on each generated style image and a style sample corresponding to each style image.

[00101] Here, during the implementation of S3002, different loss functions may be selected according to the manner the image processing model generates the style image (overall-style-based portrait stylization or style-exemplar-based portrait stylization) and a

form of processing the real sample set (picture set or video) to evaluate the total loss of the image processing model. For example, loss types may include an adversarial loss of the second generative subnetwork, a reconstruction loss of the generated style image relative to the corresponding style sample, an overlapping loss representing a consistency of overlapping regions between style images of different image frames, etc. The used loss function is not limited in the embodiment of the disclosure.

[00102] In S3003, training of the image processing model is completed in a case where the total loss satisfies the first convergence condition.

[00103] Here, after the image processing model is trained, the style image generated by the image processing model is very close to the style sample.

[00104] In the embodiment of the disclosure, each style image is generated by the image processing model, and then a total loss of each generated style image relative to the corresponding style sample is calculated, thereby implementing training of the image processing model to make the image processing model capable of meeting the convergence condition.

[00105] In the related art, a generator, such as a StyleGAN, is able to process images of a fixed size only, and requires to perform face alignment on the processed images. The image processing model provided in the embodiment of the disclosure includes a trained style encoder, a first generative subnetwork to be trained, and a trained second generative subnetwork. A first content feature generated by the first generative subnetwork is made capable for reconstruction of a structural feature of a style image, whereby first seven layers of sub-networks in the generator, such as the StyleGAN, are replaced with the first generative subnetwork. The first generative subnetwork includes a fully convolutional network. According to the translation invariance of the convolutional network, face alignment is not needed for an image processed by the image processing model. In addition, according to the reusability of a convolutional kernel in the convolutional network, the image processing model may process images of any size. Therefore, the problem in the related art that only images of a fixed size can be processed and face alignment is required is solved, the image processing model can be applied to the field of videos, the application range of the image processing model is extended, and ease of use

of the image processing model is improved.

[00106] In some embodiments, the style code includes style codes under multiple resolutions. Correspondingly, as shown in FIG. 3, the implementation of the operation in S3001 that “the image processing model is trained iteratively based on the style sample set, the real sample set, and the style code to obtain each style image generated by the image processing model” may include the following operations S31 and S32.

[00107] In S31, feature extraction is performed on the real sample set by using the first generative subnetwork to obtain third content features under different resolutions.

[00108] As described above, during the implementation of S31, feature extraction may be performed on the real sample set by using the feature extraction module in the first generative subnetwork to obtain the third content features under different resolutions.

[00109] In S32, each style image is generated by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions.

[00110] In the embodiment of the disclosure, the third content features under different resolutions are introduced to the generation of the style image, such that the generated style image includes more background information. Therefore, the generated style image is clearer, and the problem in the conventional art that the generated style image with little background information may not accurately reflect information in a video and thus the conventional art cannot be applied to videos well is solved.

[00111] In some embodiments, the third content features include content features under H resolutions. The second generative subnetwork includes K resolution layers sequentially connected from low to high resolutions. The resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions. Correspondingly, the implementation of the operation in S32 that “each style image is generated by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions” may include the following operations S321 to S324.

[00112] In S321, an output feature of a first resolution layer of the K resolution layers is generated by using the first resolution layer based on the first content feature, a style code under the first resolution layer, and the third content feature under the first resolution layer.

[00113] Here, the third content features include content features under H resolutions. For example, if the third content features include content features under $W*H$, $W/2*H/2$, $W/4*H/4$, and $W/8*H/8$, H is 4. The second generative subnetwork includes K resolution layers sequentially connected from low to high resolutions. The K resolution layers include K resolutions. Each of the resolution layers has a resolution. The resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the resolutions corresponding to the third content features. For example, the second generative subnetwork includes six resolutions layers whose resolutions are $W/8*H/8$, $W/4*H/4$, $W/2*H/2$, $W*H$, $2W*2H$, and $4W*4H$ and that are sequentially connected from low to high resolutions, and the six resolution layers include six resolutions, i.e., $W/8*H/8$, $W/4*H/4$, $W/2*H/2$, $W*H$, $2W*2H$, and $4W*4H$ respectively. Since the third content features include content features under four resolutions, the resolutions of the first four layers of the six resolution layers are in one-to-one correspondence with the four resolutions corresponding to the third content features. That is, the resolution layer whose resolution is $W/8*H/8$ corresponds to the third content feature under the resolution of $W/8*H/8$, the resolution layer whose resolution is $W/4*H/4$ corresponds to the third content feature under the resolution of $W/4*H/4$, the resolution layer whose resolution is $W/2*H/2$ corresponds to the third content feature under the resolution of $W/2*H/2$, and the resolution layer whose resolution is $W*H$ corresponds to the third content feature under the resolution of $W*H$.

[00114] A style code under the first resolution layer is the style code under the same resolution as the first resolution layer. A third content feature under the first resolution layer is the third content feature under the same resolution as the first resolution layer. S321 is for generating an output feature of the first resolution layer of the K resolution layers, namely generating an output feature of the $W/8*H/8$ resolution layer. During implementation, the first content feature, the style code under the first resolution layer, and the third content feature under the first resolution layer are input to the first

resolution layer (namely the first resolution layer includes three inputs), thereby generating the output feature of the first resolution layer.

[00115] In S322, an output feature of an $(i+1)$ -th resolution layer is generated by using the $(i+1)$ -th resolution layer based on an output feature of an i -th resolution layer, a style code under the $(i+1)$ -th resolution layer, and the third content feature under the $(i+1)$ -th resolution layer, a value of i ranging from 1 to $H-1$.

[00116] Here, a style code under the $(i+1)$ -th resolution layer is the style code under the same resolution as the $(i+1)$ -th resolution layer. A third content feature under the $(i+1)$ -th resolution layer is the third content feature under the same resolution as the $(i+1)$ -th resolution layer.

[00117] Since the resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions, S322 is for the resolution layers whose resolutions are the same as those corresponding to the third content features in the K resolution layers except the first resolution layer. Taking the condition that the third content features include features under resolutions of $W*H$, $W/2*H/2$, $W/4*H/4$, and $W/8*H/8$ and the second generative subnetwork includes resolution layers whose resolutions are $W/8*H/8$, $W/4*H/4$, $W/2*H/2$, $W*H$, $2W*2H$, and $4W*4H$ as an example for description, S322 is for generating output features of the three resolution layers of $W/4*H/4$, $W/2*H/2$, and $W*H$. During implementation, an output feature of an i -th resolution layer, a style code under an $(i+1)$ -th resolution layer, and the third content feature under the $(i+1)$ -th resolution layer are input to the $(i+1)$ -th resolution layer (namely the $(i+1)$ -th resolution layer includes three inputs), thereby generating an output feature of the $(i+1)$ -th resolution layer.

[00118] In S323, an output feature of a $(j+1)$ -th resolution layer is generated by using the $(j+1)$ -th resolution layer based on an output feature of a j -th resolution layer and a style code under the $(j+1)$ -th resolution layer, a value of j ranging from H to $K-1$.

[00119] Here, S323 is for the resolution layers whose resolutions are different from those corresponding to the third content features in the K resolution layers. Still taking the condition that the third content features include features under resolutions of $W*H$,

$W/2 \times H/2$, $W/4 \times H/4$, and $W/8 \times H/8$ and the second generative subnetwork includes resolution layers whose resolutions are $W/8 \times H/8$, $W/4 \times H/4$, $W/2 \times H/2$, $W \times H$, $2W \times 2H$, and $4W \times 4H$ as an example for description, S323 is for generating output features of the two resolution layers of $2W \times 2H$ and $4W \times 4H$. During implementation, an output feature of a j -th resolution layer and a style code under a $(j+1)$ -th resolution layer are input to the $(j+1)$ -th resolution layer (namely the $(j+1)$ -th resolution layer includes two inputs), thereby generating an output feature of the $(j+1)$ -th resolution layer.

[00120] In S324, each style image is generated based on an output feature of an H -th resolution layer.

[00121] Here, when j is valued to $K-1$ in S323, an output feature of an H -th resolution layer is generated. Still taking the condition that the third content features include features under resolutions of $W \times H$, $W/2 \times H/2$, $W/4 \times H/4$, and $W/8 \times H/8$ and the second generative subnetwork includes resolution layers whose resolutions are $W/8 \times H/8$, $W/4 \times H/4$, $W/2 \times H/2$, $W \times H$, $4W \times 2H$, and $4W \times 4H$ as an example for description, the output feature of the H -th resolution layer is an output feature of the $W/4 \times H/4$ resolution layer for generating each style image.

[00122] In the embodiment of the disclosure, for the H resolution layers in the second generative subnetwork, the first H resolution layers, compared with the latter $K-H$ resolution layers, are additionally input with the third content features under the same resolutions, so the output features generated by the first H resolution layers include more background information of the real sample. Since the output features of the first H resolution layers are input features of the latter $K-H$ resolution layers, a final output feature of the latter $K-H$ resolution layers also includes more background information of the real sample. Therefore, the generated style image includes more background information of the real sample. That is, a video image obtained by stylization includes not only a face image but also background information capable of expressing connotations of the video, such as what the person in the video is doing, where the person is, and what buildings are included in the video, all of which cannot be expressed only by the face image. Furthermore, the problem in the conventional art that the generated style image with little background information may not accurately reflect information in a video and

thus the conventional art cannot be applied to videos well is solved.

[00123] In some embodiments, each of the K resolution layers includes a generation module and a fusion module. Correspondingly, the implementation of the operation in S322 that “an output feature of an $(i+1)$ -th resolution layer is generated by using the $(i+1)$ -th resolution layer based on an output feature of an i -th resolution layer, a style code under the $(i+1)$ -th resolution layer, and the third content feature under the $(i+1)$ -th resolution layer” includes the following operations S3211 and S3212.

[00124] In S3211, a first intermediate feature of the $(i+1)$ -th resolution layer is generated by using the generation module of the $(i+1)$ -th resolution layer based on the output feature of the i -th resolution layer and the style code under the $(i+1)$ -th resolution layer.

[00125] Here, the first intermediate feature of the $(i+1)$ -th resolution layer is equivalent to an output feature of each layer in the generative network (the latter 11 layers) of the original StyleGAN. Therefore, for a method for generating the first intermediate feature of the $(i+1)$ -th resolution layer based on the output feature of the i -th resolution layer and the style code under the $(i+1)$ -th resolution layer, reference may be made to the related art in the original StyleGAN.

[00126] In S3212, the output feature of the $(i+1)$ -th resolution layer is generated by using the fusion module of the $(i+1)$ -th resolution layer based on the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer.

[00127] Here, the third content feature is a feature generated by the first generative subnetwork. Therefore, for introducing the third content feature to the second generative subnetwork, it is necessary to fuse the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer and convert them into a resolution feature that the next resolution layer may receive, i.e., the output feature of the $(i+1)$ -th resolution layer.

[00128] It is to be noted that S321 is implemented substantially by the same method as S322, and the difference is that “output feature of the i -th resolution layer” in S322 is

replaced with “first content feature” in S321. Therefore, for the implementation of S321, reference is made to S322.

[00129] In the embodiment of the disclosure, the first intermediate feature of the (i+1)-th resolution layer is generated first by the generation module of the (i+1)-th resolution layer, and then the output feature of the (i+1)-th resolution layer is generated based on the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer. In this manner, the generated output feature of the (i+1)-th resolution layer is fused with the third content feature under the (i+1)-th resolution layer, so that more background information is included.

[00130] In some embodiments, the implementation of S3212 may include the following two conditions according to the manner the image processing model generates the style image.

[00131] Under the first condition (overall-style-based portrait stylization), the implementation of S3212 may include the following operations.

[00132] In S32121, the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer are fused by using the fusion module of the (i+1)-th resolution layer to obtain the output feature of the (i+1)-th resolution layer.

[00133] Here, during the implementation of S32121, the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer are concatenated, and then are fused together through a convolutional layer to obtain the output feature of the (i+1)-th resolution layer.

[00134] In the embodiment of the disclosure, the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer are concatenated and fused, so that an (i+2)th resolution layer may receive the obtained output feature of the (i+1)-th resolution layer.

[00135] Under the second condition (style-exemplar-based portrait stylization), the implementation of S3212 may include the following operations S3212a to S3212d.

[00136] In S3212a, a difference between the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer is determined by using the fusion module of the (i+1)-th resolution layer to obtain a third feature.

[00137] Here, the implementation of S3212a may include that subtraction is performed on the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer to obtain a third feature. A portion that differs more greatly is fused at a lower probability in the subsequent process of generating the output feature of the (i+1)-th resolution layer in S3212d.

[00138] In S3212b, the third feature and the first intermediate feature of the (i+1)-th resolution layer are fused by using the fusion module of the (i+1)-th resolution layer to obtain a fourth feature.

[00139] Here, the implementation of S3212b may include that the third feature and the first intermediate feature of the (i+1)-th resolution layer are fused by a convolutional layer to obtain a fourth feature for subsequent generation of a style mask feature in combination with the style degree.

[00140] In S3212c, a style mask feature is obtained by using the fusion module of the (i+1)-th resolution layer based on the fourth feature and the style degree.

[00141] Here, the style mask feature is used to represent which portion in the third content feature needs to be fused to obtain the output feature of the (i+1)-th resolution layer and which portion of the third content feature does not need to be fused to obtain the output feature of the (i+1)-th resolution layer. For example, a portion in the style mask feature corresponding to the portion that does not need to be fused may be 0, and a portion in the style mask feature corresponding to the portion that needs to be fused may be 1. In some embodiments, the implementation of S3212c may include that the style mask feature is obtained after the fourth feature passes through an AdaIN layer taking the style degree as a condition and the convolutional layer. The style degree is used to adjust a structure of the style mask feature such that the style mask feature can represent which portion needs to be fused and which portion does not need to be fused. When the style

degree is relatively low, the style mask feature may select an element of the third content feature for fusion with the first intermediate feature as much as possible, thereby retaining details of the real sample. When the style degree is relatively high, the style mask feature selects an effective element of the third content feature for fusion with the first intermediate feature, thereby preventing ghosts.

[00142] In S3212d, the output feature of the (i+1)-th resolution layer is generated by using the fusion module of the (i+1)-th resolution layer based on the style mask feature, the first intermediate feature of the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer.

[00143] Here, the implementation of S3212d may include that the style mask feature is multiplied by the third content feature under the (i+1)-th resolution layer to obtain a fifth feature. Here, the style mask feature is used to represent which portion in the third content feature needs to be fused to obtain the output feature of the (i+1)-th resolution layer and which portion in the third content feature does not need to be fused to obtain the output feature of the (i+1)-th resolution layer, so that the obtained fifth feature may represent a feature of the portion that needs to be fused. Then, the fifth feature and the first intermediate feature of the (i+1)-th resolution layer are concatenated, and pass through a convolutional layer to obtain the output feature of the (i+1)-th resolution layer. In such case, the obtained output feature of the (i+1)-th resolution layer is a feature fused with the structural feature of the style exemplar.

[00144] In the embodiment of the disclosure, the difference between the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer is determined to obtain the third feature. Then, the third feature is fused with the first intermediate feature of the (i+1)-th resolution layer to obtain the fourth feature. Next, the fourth feature is adjusted by using the style degree to obtain the style mask feature. Since the style mask feature is used to represent which portion in the third content feature needs to be fused to obtain the output feature of the (i+1)-th resolution layer and which portion in the third content feature does not need to be fused to obtain the output feature of the (i+1)-th resolution layer, the style mask feature may be multiplied by the third content feature under the (i+1)-th resolution layer

to obtain the fifth feature for representing the portion that needs to be fused. The fifth feature is fused with the first intermediate feature of the (i+1)-th resolution layer to obtain the output feature of the (i+1)-th resolution layer fused with the structural feature of the style exemplar.

[00145] In some embodiments, the implementation of the operation in S3002 that “a total loss of the image processing model is determined based on each generated style image and a style sample corresponding to each style image” may include the following two conditions according to the manner the image processing model generates the style image.

[00146] Under the first condition (overall-style-based portrait stylization), the implementation of S3002 may include the following operation.

[00147] In S302a, an adversarial loss and a first reconstruction loss are determined respectively based on each generated style image and the style sample corresponding to each style image, and the adversarial loss and the first reconstruction loss are summed to obtain the total loss.

[00148] The adversarial loss is an adversarial loss of the second generative subnetwork. The first reconstruction loss is a loss of the generated style image relative to the corresponding style sample.

[00149] Here, an expression of a first reconstruction loss function may be:

[00150] $L_{rec} = \|\tilde{y} - y\|_2.$

[00151] y represents an unaligned style sample. $\tilde{y}$ represents the style image generated by the image processing model.

[00152] In some embodiments, another loss function for measuring a similarity between images, such as perceptual loss, may be used instead of the first reconstruction loss function.

[00153] In the embodiment of the disclosure, the adversarial loss and the first reconstruction loss are summed to obtain the total loss, thereby evaluating the

authenticity of the generated style image and the similarity between the generated style image and the corresponding style sample.

[00154] In some embodiments, a real sample in the real sample set includes an image frame set of which image frames have a sequential relationship (namely the real sample set is a set of video frames). Correspondingly, the implementation of S3002 may include the following operation.

[00155] In S302b, an adversarial loss, a first reconstruction loss and an overlapping loss are determined respectively based on each generated style image and the style sample corresponding to each style image, and the adversarial loss, the first reconstruction loss and the overlapping loss are summed to obtain the total loss.

[00156] The overlapping loss is used to represent a consistency of overlapping regions between style images of different image frames in the real sample. Here, an expression of an overlapping loss function is:

$$[00157] \quad L_{tmp} = \|\tilde{y}_1 - \tilde{y}_2\|_2.$$

[00158] $\tilde{y}_1$ and $\tilde{y}_2$ represent overlapping regions in the style image obtained by stylization in the real sample set at two viewing angles x_1 and x_2 (i.e., two image frames). Compared with S302a, the overlapping loss is added to the total loss.

[00159] In the embodiment of the disclosure, when the real sample in the real sample set includes the image frame set of which image frames have a sequential relationship, the overlapping loss is added to the total loss to evaluate the consistency of overlapping regions between the style images of different image frames in the real sample to make the overlapping regions between the style images of different image frames in the real sample as consistent as possible. Therefore, the accuracy of the image processing model is improved.

[00160] Under the second condition (style-exemplar-based portrait stylization), the implementation of S3002 may include the following operation.

[00161] In S302c, an adversarial loss, a first reconstruction loss and a mask feature

regularization loss are determined respectively based on each generated style image and the style sample corresponding to each style image, and the adversarial loss, the first reconstruction loss and the mask feature are summed to obtain the total loss.

[00162] The adversarial loss is an adversarial loss of the second generative subnetwork. The first reconstruction loss is a loss of the generated style image relative to the corresponding style sample. The mask feature regularization loss is used to constrain an average value of elements in the style mask feature. An expression of a mask feature regularization loss function is:

[00163]
$$L_{mask} = \max\{\|\mathbf{m}_E\|_1/|\mathbf{m}_E| - g(d_s), 0\}.$$

[00164] $\mathbf{m}_E$ represents the style mask feature. d_s represents the style degree. $|\mathbf{m}_E|$ represents the number of elements of $\mathbf{m}_E$. $\|\mathbf{m}_E\|_1$ represents an L2 norm. $g(d_s)$ represents a monotonic decreasing function about d_s on $[0, 1]$, and may be set to different forms as practically needed. Compared with S302a, the mask feature regularization loss is added to the total loss.

[00165] In the embodiment of the disclosure, compared with the condition of overall-style-based portrait stylization, the mask feature regularization loss is added to the total loss to evaluate the accuracy of the style mask feature. Therefore, the accuracy of the image processing model is improved.

[00166] In some embodiments, a real sample in the real sample set includes an image frame set of which image frames have a sequential relationship (namely the real sample set is a set of video frames). Correspondingly, the implementation of S3002 may include the following operation.

[00167] In S302d, an adversarial loss, a first reconstruction loss, an overlapping loss and a mask feature regularization loss are determined respectively based on each generated style image and the style sample corresponding to each style image, and the adversarial loss, the first reconstruction loss, the overlapping loss and the mask feature regularization loss are summed to obtain the total loss.

[00168] Here, for the implementation of S302d, reference is made to S302b and S302c.

The implementation of S302d provides a method for determining the total loss in a case where the real sample set is a set of video frames.

[00169] The embodiments of the disclosure also provide a StyleGAN-based portrait video stylization method. Knowledge distillation is simultaneously performed on generated data and model structure of a StyleGAN from two aspects.

[00170] At the data level, paired real face/stylized face data sets are generated by the StyleGAN-based portrait image stylization method to supervise network training without collecting real video or image data.

[00171] At the network structure level, the first seven layers of network structures of the StyleGAN for a fixed image resolution are eliminated, and a content encoder (i.e., the first generative subnetwork) is introduced to map input video frames (i.e., the real sample set) to content feature (i.e., the first content features) as an input of the eighth layer of the StyleGAN. Therefore, an end-to-end fully convolutional video stylization framework capable of supporting inputs of any resolution and processing unaligned faces is implemented.

[00172] For different StyleGAN-based portrait video stylization methods, the embodiments of the disclosure provide two specific implementation modes: collection-based portrait video stylization implemented based on a StyleGAN (i.e., overall-style-based portrait stylization) and exemplary-based portrait video stylization implemented based on DualStyleGAN (i.e., style-exemplar-based portrait stylization). The former renders a portrait to an overall style of a style data set, and the latter renders a portrait to a style in a specified exemplar image. The embodiments of the disclosure also provide a single-frame camera motion-based time-domain continuity loss function, thereby improving the continuity of video stylization.

[00173] The two specific implementation modes provided in the embodiments of the disclosure will be introduced below respectively.

[00174] Solution 1 (collection-based portrait video stylization)

[00175] 1.1. Image processing model

[00176] FIG. 4A is a composition structure diagram of an image processing model according to an embodiment of the disclosure. The image processing model includes a parameter-fixed generation model G (i.e., the above-mentioned second generative subnetwork), a parameter-fixed style encoder Es, and a content encoder E to be trained (i.e., the above-mentioned first generative subnetwork). The generation model G and the style encoder s are parameter-fixed pre-trained modules.

[00177] 1) The generation model G

[00178] A model G0 is caused to represent a StyleGAN model pre-trained based on real faces. The model G0 is finely tuned based on a style data set to obtain a StyleGAN model G1 for portrait image stylization.

[00179] The generation model G is obtained based on the model G1 by removing the first seven layers of networks and retaining only the latter 11 layers of networks, and maps a $W/8 \times H/8$ feature to a $4W \times 4H$ high-definition video frame.

[00180] 2) The style encoder Es

[00181] The style encoder Es is a pre-trained StyleGAN encoder. It maps a real sample or style exemplar to a style code of the StyleGAN, the style code being 18 512-dimensional vectors, and then inputs the latter 11 vectors to 11 layers of the generation model G for color style adjustment.

[00182] 3) The content encoder E

[00183] E includes a down-sampling convolutional layer (i.e., the feature extraction module) and a residual module (ResBlocks), and encodes an input $W \times H$ video frame into a $W/8 \times H/8$ content feature (i.e., the first content feature) for input to the generation model G.

[00184] Meanwhile, in order to retain details of the input video frame, a fusion module is added to the generation model G. A content feature f_E (i.e., the third content feature) extracted by the down-sampling convolutional layer is input to a layer of the generation model G with a corresponding resolution (i.e., the $(i+1)$ -th resolution layer or the first

resolution layer). In the present layer of the generation model G , the content feature f_E (i.e., the third content feature) of the content encoder and the feature f_G (i.e., the first intermediate feature) of the generation model are concatenated by channel, and then pass through a convolutional layer to obtain a fused feature f_{out} (i.e., the output feature of the $(i+1)$ -th resolution layer or the first resolution layer) as an input of a next resolution layer of the generation model G . For $2W*2H$ and $4W*4H$ layers (i.e., the $(j+1)$ -th resolution layer) in the generation model, a style code under the corresponding resolution and an output features of a previous resolution layer are input, thereby obtaining an output feature of the $(j+1)$ -th resolution layer.

[00185] 1.2. Training scheme

[00186] In a training stage, training is performed fixedly based on images satisfying $W=256$ and $H=256$.

[00187] 1.2.1. Training data

[00188] A data set for training is generated by the complete StyleGAN model G_0 and model G_1 . A randomly sampled Gaussian noise z is used to generate a real portrait image $x'=G_0(z)$ and a corresponding artistic portrait image $y'=G_1(z)$. The same geometric-transformation-based data enhancement is performed on x' and y' to obtain unaligned data (x, y) , i.e., the real sample and the style sample.

[00189] 1.2.2. Model initialization

[00190] In a first aspect, the pre-trained content encoder E simulates output features of the first seven layers of subnetworks of the model G_1 . This feature is set to f_{G_1} (i.e., the second content feature). An output feature of the last layer of the content encoder E is f'_E (i.e., the first content feature). The real portrait image generated based on z is $x'=G_0(z)$. Then, a target equation for pre-training is:

$$[00191] \quad L_E = \|\mathbf{f}'_E(x'_\downarrow) - \mathbf{f}_{G_1}(z)\|.$$

[00192] $\downarrow$ represents performing quadruple down-sampling on the image so as to meet an input resolution requirement of E . Since resolutions of images generated by the model

G1 and the model G0 are 1024*1024, while a resolution of an image input to the style encoder is required to be 256*256, and images input to the style encoder E_s and the content encoder E are the same image, quadruple down-sampling should be performed on the real portrait image x' generated by the model G0, the resulting image is input to the style encoder E_s and the content encoder E .

[00193] In a second aspect, a portion, interacting with the content feature f_E (i.e., the third content feature), of a convolutional kernel of a convolutional layer in the fusion module is initialized to an approximately all-0 matrix, and a portion interacting with the feature f_G of the generation model is initialized to a unit matrix. In this manner, the output by the fusion module at the beginning of training (i.e., the output feature of the (i+1)-th resolution layer or the first resolution layer) is approximately equal to f_G , namely the generation model G is not fused with the content feature f_E (i.e., the third content feature) generated by the content encoder. Therefore, the model initialization may simulate the model G1 to generate a rough stylized portrait.

[00194] 1.2.3. Model training

[00195] The purpose is to train the network to map x to y , and a reconstruction loss L_{rec} is used. During implementation, the network is caused to obtain a stylization result $\tilde{y} = G(E(x_l), E_s(x'_l))$ according to multi-scale content features $E(x_l)$ (i.e., content features f_E) and style codes $E_s(x'_l)$, and then an expression of a reconstruction loss function is:

[00196]
$$L_{rec} = \|\tilde{y} - y\|_2.$$

[00197] In some embodiments, besides an L2 regularization loss function, another loss function for measuring a similarity between images may be used, such as perceptual loss. In addition, an adversarial loss L_{adv} may also be used to improve the authenticity of $\tilde{y}$.

[00198] In addition, for the time-domain continuity of the video, FIG. 4B is a schematic diagram of overlapping regions between style images of different image frames in a real sample. For a single frame of image, a translation motion of a camera is simulated, corresponding to two viewing angles x_1 and x_2 (i.e., two image frames) of the

video frame, and overlapping regions $\tilde{y}_1$ and $\tilde{y}_2$ of style images obtained by stylization should be consistent. In such case, an overlapping loss is used to evaluate a consistency of the overlapping regions between the style images of different image frames. Correspondingly, an expression of an overlapping loss function is:

[00199]
$$L_{tmp} = \|\tilde{y}_1 - \tilde{y}_2\|_2.$$

[00200] In summary, a total loss of the image processing model is $L_{rec} + L_{adv} + L_{tmp}$.

[00201] 1.3. Implementation solution

[00202] In practical applications, for a W*H video, face detection is performed on a first frame of the video, and a 256*256 face region is obtained by cropping and input to the style encoder Es to extract a style code that can be applied to all of subsequent frames. Alternatively, a style code of each frame may be extracted independently. A 4W*4H high-resolution stylized video frame is generated by the generation model G based on multi-scale content features $E(x_{\downarrow})$ extracted by inputting each frame to the content encoder E. All the stylized video frames form a final stylized portrait video.

[00203] Solution 2 (exemplar-based portrait video stylization)

[00204] 2.1. Image processing model

[00205] FIG. 5A is a composition structure diagram of another image processing model according to an embodiment of the disclosure. The image processing model is substantially the same as collection-based portrait video stylization, and the difference is that the generation model G uses the latter 11 layers of networks and first seven layers of ModRes modules (i.e., residual adjustment module) of the DualStyleGAN pre-trained based on the style data set.

[00206] The ModRes module is configured to adjust a feature (i.e., the second intermediate feature) generated by the residual module in the content encoder E. Since resolutions of features of the first seven layers of DualStyleGAN are different from that of the feature generated by the residual module, a convolutional layer of the ModRes module is replaced with atrous convolution, with a convolutional kernel remaining

unchanged, to match a receptive field by changing an expansion rate. For example, an expansion rate of the ModRes module corresponding to the feature whose resolution is $8*8$ in the features of the first seven layers is set to 4, and then the ModRes module may be configured to process a $32*32$ feature generated by the residual module.

[00207] Like DualStyleGAN, the input of the network includes not only the input video frame but also an additional style exemplar and style degree ds . The content encoder E encodes the video frame into multi-scale content features f_E (i.e., the third content features). The content feature f_E is adjusted by the ModRes module according to the degree of ds , and then is input to the generation model G . The style encoder E_s maps the style exemplar to a style code of the StyleGAN, with the first seven vectors being input to the ModRes module and the latter 11 vectors being input to the generation model G . The generation model G maps the $W/8*H/8$ feature to a $4W*4H$ high-definition video frame according to the multi-scale content features and the style code.

[00208] The fusion module calculates a difference between the content feature f_E of the content encoder and the feature f_G of the generation model first. The f_E is concatenated with f_G , passes through the convolutional layer, and then passes through an AdaIN layer taking ds as a condition and the convolutional layer, thereby obtaining a mask $\mathbf{m}_E$ (i.e., the style mask feature). $\mathbf{m}_E$ is multiplied by f_E and then concatenated with f_G by channel, and then pass through a convolutional layer, thereby obtaining a fused feature f_{out} (i.e., the output feature of the $(i+1)$ -th resolution layer or the first resolution layer). When ds is relatively low, the mask selects an element of f_E for fusion with f_G as much as possible, thereby retaining details of the video frame. When ds is relatively high, the mask selects an effective element of f_E for fusion with f_G , thereby preventing ghosts.

[00209] 2.2. Training scheme

[00210] In a training stage, training is performed fixedly based on images satisfying $W=256$ and $H=256$.

[00211] 2.2.1. Training data

[00212] A data set for training is generated by the complete DualStyleGAN model $G1$. A randomly sampled Gaussian noise z , a style exemplar S , and the style degree ds are

used to generate a real portrait image $x'=G1(z, Es(S), 0)$ and a corresponding artistic portrait image $y'=G1(z, Es(S), ds)$. In case of $ds=0$, a face image that is not stylized is output. The same geometric-transformation-based data enhancement is performed on x' and y' to obtain unaligned data (x, y) , i.e., the real sample and the style sample.

[00213] 2.2.2. Model initialization

[00214] The model initialization is substantially the same as collection-based portrait video stylization. A target equation for pre-training is:

$$[00215] \quad L_E = \|\mathbf{f}'_E(x'_\downarrow, E_s(S), d_s) - \mathbf{f}_{G_1}(z, E_s(S), d_s)\|.$$

[00216] $E_s(S)$ and ds are additionally input to E and $G1$.

[00217] 2.2.3. Model training

[00218] The model training is substantially the same as collection-based portrait video stylization, and a loss function (i.e., the mask feature regularization loss) L_{mask} for constraining the mask is added additionally.

$$[00219] \quad L_{mask} = \max\{\|\mathbf{m}_E\|_1/|\mathbf{m}_E| - g(d_s), 0\}.$$

[00220] $\mathbf{m}_E$ represents a style mask feature. ds represents the style degree. $|\mathbf{m}_E|$ represents the number of elements of $\mathbf{m}_E$. $\|\mathbf{m}_E\|_1$ represents an L2 norm. $g(d_s)$ represents a monotonic decreasing function about ds on $[0, 1]$, and may be set to different forms as practically needed.

[00221] 2.3. Implementation solution

[00222] The implementation solution is substantially the same as collection-based portrait video stylization, and the style exemplar S and the style degree ds are additionally added as inputs.

[00223] The embodiments of the disclosure provide an image processing method, which is applied to an image processing model. The image processing model includes a style encoder, a first generative subnetwork and a second generative subnetwork. The

style encoder is configured to map a style exemplar or content image to a style code. The first generative subnetwork includes a fully convolutional network, and is configured to generate a first content feature of the content image. The second generative subnetwork is configured to generate a style image of the content image based on the style code and the first content feature. As shown in FIG. 5B, the method at least includes the following operations S401, S402, and S403.

[00224] In S401, the style encoder generates the style code of the content image, the content image being an image determined from an image frame set to be processed, and the image frame set having image frames with a sequential relationship.

[00225] Here, the image frame set to be processed may be a set of video frames. In some embodiments, the image frame set may also be an image set without a sequential relationship.

[00226] The method may be implemented as a computer-executable program, embodied as a client at a user side. A user selects the image frame set to be processed. For example, the user locally selects a video. The client receives a selection operation of the user. That is, the client determines the selected video as the image frame set to be processed. The content image is any image determined from the image frame set to be processed.

[00227] The style code may be generated by inputting the content image or the style exemplar to the style encoder.

[00228] In S402, the first generative subnetwork generates, based on the content image, the first content feature capable for reconstruction of a structural feature of the style image.

[00229] During implementation, the content image is input to the first generative subnetwork to generate the first content feature capable for reconstruction of the structural feature of the style image.

[00230] In S403, the second generative subnetwork generates the style image based on the first content feature and the style code.

[00231] During implementation, the first content feature and the style code are input to the second generative subnetwork to generate the style image. The style code may be a color style code.

[00232] In some embodiments, the style code includes style codes under different resolutions. The operation that the second generative subnetwork generates the style image based on the first content feature and the style code includes the following operations. Feature extraction is performed on the content image by using the first generative subnetwork to obtain third content features under different resolutions. The style image is generated by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions.

[00233] In some embodiments, the third content features include content features under H resolutions. The second generative subnetwork includes K resolution layers sequentially connected from low to high resolutions. The resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions. The operation that each style image is generated by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions includes the following operations. An output feature of a first resolution layer of the K resolution layers is generated by using the first resolution layer based on the first content feature, a style code under the first resolution layer, and the third content feature under the first resolution layer. An output feature of an (i+1)-th resolution layer is generated by using the (i+1)-th resolution layer based on an output feature of an i-th resolution layer, a style code under the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer, a value of i ranging from 1 to H-1. An output feature of a (j+1)-th resolution layer is generated by using the (j+1)-th resolution layer based on an output feature of a j-th resolution layer and a style code under the (j+1)-th resolution layer, a value of j ranging from H to K-1. The style image is generated based on an output feature of an H-th resolution layer.

[00234] In some embodiments, the operation that an output feature of an (i+1)-th resolution layer is generated by using the (i+1)-th resolution layer based on an output

feature of an i -th resolution layer, a style code under the $(i+1)$ -th resolution layer, and the third content feature under the $(i+1)$ -th resolution layer includes the following operations. A first intermediate feature of the $(i+1)$ -th resolution layer is generated by using the $(i+1)$ -th resolution layer based on the output feature of the i -th resolution layer and the style code under the $(i+1)$ -th resolution layer. The output feature of the $(i+1)$ -th resolution layer is generated by using the $(i+1)$ -th resolution layer based on the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer.

[00235] In some embodiments, the first generative subnetwork includes a feature extraction module, a residual module, and a residual adjustment module. The operation that the first generative subnetwork generates, based on the content image, the first content feature capable for reconstruction of a structural feature of the style image includes the following operations. A structural style code of the content image is acquired by using the style encoder based on the style exemplar. A style degree of the style exemplar is acquired. Feature extraction is performed on the content image by using the feature extraction module to obtain the third content features under different resolutions. Feature extraction is performed on the third content feature under a minimum resolution of the different resolutions by using the residual module to obtain a second intermediate feature. The second intermediate feature is adjusted by using the residual adjustment module based on the style degree and the structural style code to generate the first content feature.

[00236] In some embodiments, the operation that the output feature of the $(i+1)$ -th resolution layer is generated by using the $(i+1)$ -th resolution layer based on the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer includes the following operations. A difference between the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer is determined by using the $(i+1)$ -th resolution layer to obtain a third feature. The third feature and the first intermediate feature of the $(i+1)$ -th resolution layer are fused by using the second generative subnetwork to obtain a fourth feature. A style mask feature is obtained by using the second generative subnetwork based on the fourth feature and the style degree. The output feature of the $(i+1)$ -th

resolution layer is generated by using the second generative subnetwork based on the style mask feature, the first intermediate feature of the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer.

[00237] Based on the embodiments, the embodiments of the disclosure provide apparatus for training an image processing model. Each unit of the apparatus and each module of each unit may be implemented by a processor in a computer device, or may certainly be implemented by a specific logic circuit. During implementation, the processor may be a Central Processing Unit (CPU), a Microprocessor Unit (MPU), a Digital Signal Processor (DSP), a Field Programmable Gate Array (FPGA), or the like.

[00238] FIG. 6 is a composition structure diagram of an apparatus for training an image processing model according to an embodiment of the disclosure. An image processing model includes a style encoder, a first generative subnetwork and a second generative subnetwork. The style encoder is configured to map a style exemplar or real sample to a style code. The first generative subnetwork includes a fully convolutional network, and is configured to generate a first content feature of the real sample. The second generative subnetwork is configured to generate a style image of the real sample based on the style code and the first content feature. As shown in FIG. 6, the apparatus 600 for training an image processing model includes a first acquisition module 610, a first training module 620, and a second training module 630.

[00239] The first acquisition module 610 is configured to acquire a style sample set and a real sample set.

[00240] The first training module 620 is configured to train the first generative subnetwork based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image.

[00241] The second training module 630 is configured to, after training of the first generative subnetwork completes, train the image processing model iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied.

- [00242] In some embodiments, the first training module 620 includes a first generation submodule, a second generation submodule, a third generation submodule, a first determination submodule, and a first training submodule, which are configured to implement S2001 to S2005 respectively.
- [00243] In some embodiments, the second training module 630 includes a second training submodule, a second determination submodule, and a completion submodule, which are configured to implement S3001 to S3003 respectively.
- [00244] In some embodiments, the style code includes style codes under multiple resolutions. The second training submodule includes a first extraction unit and a first generation unit, which are configured to implement S31 to S32 respectively.
- [00245] In some embodiments, the third content features include content features under H resolutions. The second generative subnetwork includes K resolution layers sequentially connected from low to high resolutions. The resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions. The first generation unit includes a first generation subunit, a second generation subunit, a third generation subunit, and a fourth generation subunit, which are configured to implement S321 to S324 respectively.
- [00246] In some embodiments, each of the K resolution layers includes a generation module and a fusion module. The second generation subunit is further configured to implement S3211 and S3212.
- [00247] In some embodiments, the second generation subunit is further configured to implement S32121.
- [00248] A real sample in the real sample set includes an image frame set of which image frames have a sequential relationship. The second determination submodule is further configured to implement S302b.
- [00249] In some embodiments, the first generative subnetwork includes a feature extraction module, a residual module, and a residual adjustment module. The third generation submodule includes an acquisition unit, a second extraction unit, a third

extraction unit, and a third generation unit, which are configured to implement S223a to S223d respectively.

[00250] In some embodiments, the first generation subunit is further configured to implement S3212a to S3212d.

[00251] In some embodiments, the second determination submodule is further configured to implement S302c.

[00252] FIG. 7 is a composition structure diagram of an image processing apparatus according to an embodiment of the disclosure. The apparatus is applied to an image processing model. The image processing model includes a style encoder, a first generative subnetwork and a second generative subnetwork. The style encoder is configured to map a style exemplar or content image to a style code. The first generative subnetwork includes a fully convolutional network, and is configured to generate a first content feature of the content image. The second generative subnetwork is configured to generate a style image of the content image based on the style code and the first content feature. The image processing apparatus 700 includes a first generation module 710, a reconstruction module 720, and a second generation module 730.

[00253] The first generation module 710 is configured to generate the style code of the content image by using the style encoder, the content image being an image determined from an image frame set to be processed, and the image frame set having image frames with a sequential relationship.

[00254] The reconstruction module 720 is configured to generate, based on the content image by using the first generative subnetwork, the first content feature capable for reconstruction of a structural feature of the style image.

[00255] The second generation module 730 is configured to generate the style image by using the second generative subnetwork based on the first content feature and the style code.

[00256] In some embodiments, the style code includes style codes under different resolutions. The second generation module 730 includes a first extraction submodule and

a fourth generation submodule. The first extraction submodule is configured to perform feature extraction on the content image by using the first generative subnetwork to obtain third content features under different resolutions. The fourth generation submodule is configured to generate the style image by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions.

[00257] In some embodiments, the third content features include content features under H resolutions. The second generative subnetwork includes K resolution layers sequentially connected from low to high resolutions. The resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions. The fourth generation submodule includes a fourth generation unit, a fifth generation unit, a sixth generation unit, and a seventh generation unit. The fourth generation unit is configured to generate an output feature of a first resolution layer of the K resolution layers by using the first resolution layer based on the first content feature, a style code under the first resolution layer, and the third content feature under the first resolution layer. The fifth generation unit is configured to generate an output feature of an $(i+1)$ -th resolution layer by using the $(i+1)$ -th resolution layer based on an output feature of an i -th resolution layer, a style code under the $(i+1)$ -th resolution layer, and the third content feature under the $(i+1)$ -th resolution layer, a value of i ranging from 1 to $H-1$. The sixth generation unit is configured to generate an output feature of a $(j+1)$ -th resolution layer by using the $(j+1)$ -th resolution layer based on an output feature of a j -th resolution layer and a style code under the $(j+1)$ -th resolution layer, a value of j ranging from H to $K-1$. The seventh generation unit is configured to generate the style image based on an output feature of an H -th resolution layer.

[00258] In some embodiments, the fifth generation unit includes a fifth generation subunit and a sixth generation subunit. The fifth generation subunit is configured to generate a first intermediate feature of the $(i+1)$ -th resolution layer by using the $(i+1)$ -th resolution layer based on the output feature of the i -th resolution layer and the style code under the $(i+1)$ -th resolution layer. The sixth generation subunit is configured to generate the output feature of the $(i+1)$ -th resolution layer by using the $(i+1)$ -th resolution layer based on the first intermediate feature of the $(i+1)$ -th resolution layer and the third

content feature under the (i+1)-th resolution layer.

[00259] In some embodiments, the first generative subnetwork includes a feature extraction module, a residual module, and a residual adjustment module. The reconstruction module includes a first acquisition submodule, a second acquisition submodule, a second extraction submodule, a third extraction submodule, and an adjustment submodule. The first acquisition submodule is configured to acquire a structural style code of the content image is acquired by using the style encoder based on the style exemplar. The second acquisition submodule is configured to acquire a style degree of the style exemplar. The second extraction submodule is configured to perform feature extraction on the content image by using the feature extraction module to obtain the third content features under different resolutions. The third extraction submodule is configured to perform feature extraction on the third content feature under a minimum resolution of the different resolutions by using the residual module to obtain a second intermediate feature. The adjustment submodule is configured to adjust the second intermediate feature by using the residual adjustment module based on the style degree and the structural style code to generate the first content feature.

[00260] In some embodiments, the sixth generation subunit is further configured to determine a difference between the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer is determined by using the (i+1)-th resolution layer to obtain a third feature, fuse the third feature and the first intermediate feature of the (i+1)-th resolution layer by using the (i+1)-th resolution layer to obtain a fourth feature, obtain a style mask feature by using the (i+1)-th resolution layer based on the fourth feature and the style degree, and generate the output feature of the (i+1)-th resolution layer by using the (i+1)-th resolution layer based on the style mask feature, the first intermediate feature of the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer.

[00261] The descriptions about the above apparatus embodiment are similar to those about the method embodiment, and beneficial effects similar to those of the method embodiment are achieved. In some embodiments, functions or modules of the apparatus provided in the embodiment of the disclosure may be configured to execute the method

described in the method embodiment, and technical details undisclosed in the apparatus embodiment of the disclosure are understood with reference to the descriptions about the method embodiment of the disclosure.

[00262] If the technical solution of the disclosure involves personal information, a product using the technical solution of the disclosure has clearly informed personal information processing rules and obtained an independent consent of an individual before processing the personal information. If the technical solution of the disclosure involves sensitive personal information, a product using the technical solution of the disclosure has obtained an independent consent of an individual and simultaneously met a requirement for “express consent” before processing the sensitive personal information. For example, a clear and conspicuous identifier is set at a personal information acquisition apparatus such as a camera to indicate that a personal information acquisition range has been entered and personal information is to be acquired, and if an individual volunteers to enter the acquisition range, it indicates that the individual consents his/her personal information to be acquired. Alternatively, when personal information processing rules are informed by using a conspicuous identifier/information on a personal information processing apparatus, an individual mandate is obtained by pop-up information, or asking an individual to upload his/her personal information independently, or other manners. The personal information processing rules may include a personal information processor, a personal information processing purpose, a processing manner, a type of processed personal information, and other information.

[00263] It is to be noted that, in the embodiments of the disclosure, when being implemented in form of a software function module and sold or used as an independent product, the image processing method or the method for training an image processing model may also be stored in a computer-readable storage medium. Based on such an understanding, the technical solutions of the embodiments of the disclosure substantially or parts making contributions to the related art may be embodied in form of a software product. The software product is stored in a storage medium, including a plurality of instructions for causing a computer device (which may be a personal computer, a server, a network device, or the like) to execute all or part of the method in each embodiment of the disclosure. The storage medium includes various media capable of storing program

codes, such as a U disk, a mobile hard disk, a Read Only Memory (ROM), a magnetic disk, or an optical disk. Therefore, the embodiments of the disclosure are not limited to any specific hardware, software, or firmware, or any combination of hardware, software, and firmware.

[00264] The embodiments of the disclosure provide a computer device, which includes a memory and a processor. The memory stores a computer program capable of running in the processor. The processor executes the program to implement part or all of the steps in the method.

[00265] The embodiments of the disclosure provide a computer-readable storage medium, which stores a computer program executed by a processor to implement part or all of the steps in the method. The computer-readable storage medium may be transitory or non-transitory.

[00266] The embodiments of the disclosure provide a computer program, which includes a computer-readable code. Under the condition that the computer-readable code runs in a computer device, a processor of the computer device executes part or all of the steps in the method.

[00267] The embodiments of the disclosure provide a computer program product, which includes a non-transitory computer-readable storage medium storing a computer program. The computer program is read and executed by a computer to implement part or all of the steps in the method. The computer program product may specifically be implemented by hardware, software, or a combination thereof. In some embodiments, the computer program product is specifically embodied as a computer storage medium. In some other embodiments, the computer program product is specifically embodied as a software product, such as a Software Development Kit (SDK).

[00268] It is to be pointed out that the above descriptions about the embodiments focus on differences between each embodiment and the same or similar parts may refer to each other. The above descriptions about the device, storage medium, computer program, and computer program product embodiments are similar to those about the method embodiments, and beneficial effects similar to those of the method embodiments are

achieved. Technical details undisclosed in the device, storage medium, computer program, and computer program product embodiments of the disclosure are understood with reference to the descriptions about the method embodiments of the disclosure.

[00269] It is to be noted that FIG. 8 is a schematic diagram of a hardware entity of a computer device according to an embodiment of the disclosure. As shown in FIG. 8, the hardware entity of the computer device 800 includes a processor 801, a communication interface 802, and a memory 803. The processor 801 usually controls overall operations of the computer device 800. The communication interface 802 may enable the computer device to communicate with another terminal or server through a network.

[00270] The memory 803 is configured to store an instruction and application executable for the processor 801, may also cache data (for example, image data, video data, voice communication data, and video communication data) to be processed or having been processed by the processor 801 and each module in the computer device 800, and may be implemented through a flash or a Random Access Memory (RAM). The processor 801, the communication interface 802, and the memory 803 may perform data transmission with one another through a bus 804.

[00271] It is to be understood that "one embodiment" and "an embodiment" mentioned in the whole specification mean that specific features, structures, or characteristics related to the embodiment are included in at least one embodiment of the disclosure. Therefore, "in one embodiment" or "in an embodiment" mentioned throughout the specification does not necessarily refer to the same embodiment. In addition, these specific features, structures, or characteristics may be combined in one or more embodiments as appropriate. It is to be understood that, in each embodiment of the disclosure, the magnitude of the sequence number of each step/process does not mean an execution sequence, and the execution sequence of each step/process should be determined by its function and an internal logic and should not form any limitation on the implementation of the embodiments of the disclosure. The sequence numbers of the embodiments of the disclosure are only for description and do not represent superiority-inferiority of the embodiments.

[00272] It is to be noted that terms "include" and "contain" or any other variation

thereof is intended to cover nonexclusive inclusions herein, so that a process, method, object, or device including a series of elements not only includes those elements but also includes other elements which are not clearly listed, or further includes elements intrinsic to the process, the method, the object, or the device. With no more restrictions, an element defined by statement "including a/an....." does not exclude the existence of the same other elements in a process, method, object, or device including the element.

[00273] In some embodiments provided by the disclosure, it is to be understood that the disclosed device and method may be implemented in another manner. The device embodiment described above is only schematic. For example, the division of the units is only logic function division, and other division manners may be used during practical implementation. For example, multiple units or components may be combined or integrated into another system, or some characteristics may be neglected or not executed. In addition, coupling or direct coupling or communication connection between each displayed or discussed component may be indirect coupling or communication connection, implemented through some interfaces, of the device or the units, and may be electrical or mechanical or use other forms.

[00274] The units described as separate parts may or may not be physically separated. Parts displayed as units may or may not be physical units. That is, they are located in the same place or distributed to multiple network units. Part or all of the units may be selected as practically required to achieve the objectives of the solutions of the embodiments.

[00275] In addition, each function unit in each embodiment of the disclosure may be integrated into a processing unit. Alternatively, each unit may serve as an independent unit. Alternatively, two or more than two units may be integrated into a unit. The integrated unit may be implemented in a hardware form or in form of a hardware and software function unit.

[00276] It can be understood by those of ordinary skill in the art that all or part of the steps of the method embodiment may be implemented by related hardware instructed through a program. The program may be stored in a computer-readable storage medium. When the program is executed, the steps of the method embodiment are executed. The

storage medium includes various media capable of storing program codes, such as a mobile storage device, a ROM, a magnetic disk, or an optical disk.

[00277] Alternatively, the integrated unit of the disclosure may be stored in a computer-readable storage medium when being implemented in form of a software function module and sold or used as an independent product. Based on such an understanding, the technical solutions of the disclosure substantially or parts making contributions to the related art may be embodied in form of a software product. The computer software product is stored in a storage medium, including a plurality of instructions for causing a computer device (which may be a personal computer, a server, a network device, or the like) to execute all or part of the method in each embodiment of the disclosure. The storage medium includes various media capable of storing program codes, such as a mobile hard disk, a ROM, a magnetic disk, or an optical disk.

[00278] The above is only the implementation mode of the disclosure and not intended to limit the scope of protection of the disclosure. Any variations or replacements apparent to those skilled in the art within the technical scope disclosed by the disclosure shall fall within the scope of protection of the disclosure.

CLAIMS

1. An image processing method, applied to an image processing model, the image processing model comprising a style encoder, a first generative subnetwork and a second generative subnetwork, wherein the style encoder is configured to map a style exemplar or content image to a style code, the first generative subnetwork comprises a fully convolutional network, and is configured to generate a first content feature of the content image, and the second generative subnetwork is configured to generate a style image of the content image based on the style code and the first content feature, the method comprising:

generating, by the style encoder, the style code of the content image, the content image being an image determined from an image frame set to be processed, and the image frame set having image frames with a sequential relationship;

generating, by the first generative subnetwork based on the content image, the first content feature capable for reconstruction of a structural feature of the style image; and

generating, by the second generative subnetwork, the style image based on the first content feature and the style code.

2. The method of claim 1, wherein the style code comprises style codes under different resolutions; and the generating, by the second generative subnetwork, the style image based on the first content feature and the style code comprises:

performing feature extraction on the content image by using the first generative subnetwork to obtain third content features under different resolutions, and

generating the style image by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions.

3. The method of claim 2, wherein the third content features comprise content features under H resolutions, the second generative subnetwork comprises K resolution layers sequentially connected from low to high resolutions, and the resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions; and

the generating the style image by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions comprises:

generating an output feature of a first resolution layer of the K resolution layers by using the first resolution layer based on the first content feature, a style code under the first resolution layer, and the third content feature under the first resolution layer,

generating an output feature of an (i+1)-th resolution layer by using the (i+1)-th resolution layer based on an output feature of an i-th resolution layer, a style code under the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer, a value of i ranging from 1 to H-1,

generating an output feature of a (j+1)-th resolution layer by using the (j+1)-th resolution layer based on an output feature of a j-th resolution layer and a style code under the (j+1)-th resolution layer, a value of j ranging from H to K-1, and

generating the style image based on an output feature of an H-th resolution layer.

4. The method of claim 3, wherein the generating an output feature of an (i+1)-th resolution layer by using the (i+1)-th resolution layer based on an output feature of an i-th resolution layer, a style code under the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer comprises:

generating a first intermediate feature of the (i+1)-th resolution layer by using the (i+1)-th resolution layer based on the output feature of the i-th resolution layer and the style code under the (i+1)-th resolution layer; and

generating the output feature of the (i+1)-th resolution layer by using the (i+1)-th

resolution layer based on the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer.

5. The method of claim 4, wherein the first generative subnetwork comprises a feature extraction module, a residual module, and a residual adjustment module; and the generating, by the first generative subnetwork based on the content image, the first content feature capable for reconstruction of a structural feature of the style image comprises:

acquiring a structural style code of the content image by using the style encoder based on the style exemplar;

acquiring a style degree of the style exemplar;

performing feature extraction on the content image by using the feature extraction module to obtain the third content features under different resolutions;

performing feature extraction on the third content feature under a minimum resolution of the different resolutions by using the residual module to obtain a second intermediate feature; and

adjusting the second intermediate feature by using the residual adjustment module based on the style degree and the structural style code to generate the first content feature.

6. The method of claim 5, wherein the generating the output feature of the (i+1)-th resolution layer by using the (i+1)-th resolution layer based on the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer comprises:

determining a difference between the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer by using the (i+1)-th resolution layer to obtain a third feature;

fusing the third feature and the first intermediate feature of the (i+1)-th resolution layer by using the (i+1)-th resolution layer to obtain a fourth feature;

obtaining a style mask feature by using the (i+1)-th resolution layer based on the fourth feature and the style degree; and

generating the output feature of the (i+1)-th resolution layer by using the (i+1)-th resolution layer based on the style mask feature, the first intermediate feature of the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer.

7. A method for training an image processing model comprising a trained style encoder, a first generative subnetwork to be trained and a trained second generative subnetwork, wherein the style encoder is configured to map a style exemplar or real sample into a style code, the first generative subnetwork comprises a fully convolutional network, and is configured to generate a first content feature of the real sample, and the second generative subnetwork is configured to generate a style image of the real sample based on the style code and the first content feature,

the method comprising:

acquiring a style sample set and a real sample set;

training the first generative subnetwork based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image; and

after training of the first generative subnetwork completes, training the image processing model iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied.

8. The method of claim 7, wherein the training the first generative subnetwork based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image comprises:

generating the style code by using the style encoder, the style code comprising a

structural style code and a color style code;

generating a second content feature by using a third generative subnetwork of a generator based on the structural style code and the real sample set, the generator comprising the style encoder, the third generative subnetwork and the second generative subnetwork, and the third generative subnetwork being configured to affect a structural feature of a style image corresponding to a real sample of a fixed size;

generating the first content feature by using the first generative subnetwork based on the real sample set;

determining a first regularization loss between each first content feature and the corresponding second content feature, the first regularization loss being used to represent a difference between the first content feature and the second content feature; and

training the first generative subnetwork iteratively based on the first regularization loss until a second convergence condition is satisfied.

9. The method of claim 8, wherein the training the image processing model iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied comprises:

training the image processing model iteratively based on the style sample set, the real sample set, and the style code to obtain each style image generated by the image processing model;

determining a total loss of the image processing model based on each generated style image and a style sample corresponding to each style image; and

completing training of the image processing model in a case where the total loss satisfies the first convergence condition.

10. The method of claim 9, wherein the style code comprises style codes under multiple

resolutions; and the training the image processing model iteratively based on the style sample set, the real sample set, and the style code to obtain each style image generated by the image processing model comprises:

performing feature extraction on the real sample set by using the first generative subnetwork to obtain third content features under different resolutions; and

generating each style image by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions.

11. The method of claim 10, wherein the third content features comprise content features under H resolutions, the second generative subnetwork comprises K resolution layers sequentially connected from low to high resolutions, and the resolutions of the first H layers of the K resolution layers are in one-to-one correspondence with the H resolutions; and

the generating each style image by using the second generative subnetwork based on the first content feature, the third content features under different resolutions, and the style codes under different resolutions comprises:

generating an output feature of a first resolution layer of the K resolution layers by using the first resolution layer based on the first content feature, a style code under the first resolution layer, and the third content feature under the first resolution layer,

generating an output feature of an (i+1)-th resolution layer by using the (i+1)-th resolution layer based on an output feature of an i-th resolution layer, a style code under the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer, a value of i ranging from 1 to H-1,

generating an output feature of a (j+1)-th resolution layer by using the (j+1)-th resolution layer based on an output feature of a j-th resolution layer and a style code under the (j+1)-th resolution layer, a value of j ranging from H to K-1, and

generating each style image based on an output feature of an H-th resolution layer.

12. The method of claim 11, wherein each of the K resolution layers comprises a generation module and a fusion module; and the generating an output feature of an $(i+1)$ -th resolution layer by using the $(i+1)$ -th resolution layer based on an output feature of an i -th resolution layer, a style code under the $(i+1)$ -th resolution layer, and the third content feature under the $(i+1)$ -th resolution layer comprises:

generating a first intermediate feature of the $(i+1)$ -th resolution layer by using the generation module of the $(i+1)$ -th resolution layer based on the output feature of the i -th resolution layer and the style code under the $(i+1)$ -th resolution layer, and

generating the output feature of the $(i+1)$ -th resolution layer by using the fusion module of the $(i+1)$ -th resolution layer based on the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer.

13. The method of claim 12, wherein the generating the output feature of the $(i+1)$ -th resolution layer by using the fusion module of the $(i+1)$ -th resolution layer based on the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer comprises:

fusing the first intermediate feature of the $(i+1)$ -th resolution layer and the third content feature under the $(i+1)$ -th resolution layer by using the fusion module of the $(i+1)$ -th resolution layer to obtain the output feature of the $(i+1)$ -th resolution layer;

a real sample in the real sample set comprises an image frame set of which image frames have a sequential relationship; and correspondingly, the determining a total loss of the image processing model based on each generated style image and a style sample corresponding to each style image comprises:

determining an adversarial loss, a first reconstruction loss and an overlapping loss respectively based on each generated style image and the style sample corresponding to each style image, and summing the adversarial loss, the first reconstruction loss and the

overlapping loss to obtain the total loss,

wherein the adversarial loss is an adversarial loss of the second generative subnetwork, the first reconstruction loss is a loss of the generated style image relative to the corresponding style sample, and the overlapping loss is used to represent a consistency of overlapping regions between style images of different image frames in the real sample.

14. The method of claim 12, wherein the first generative subnetwork comprises a feature extraction module, a residual module, and a residual adjustment module; and the generating the first content feature by using the first generative subnetwork based on the real sample set comprises:

acquiring a style degree of the style exemplar;

performing feature extraction on the real sample set by using the feature extraction module to obtain the third content features under different resolutions;

performing feature extraction on the third content feature under a minimum resolution of the different resolutions by using the residual module to obtain a second intermediate feature; and

adjusting the second intermediate feature by using the residual adjustment module based on the style degree and the style code to generate the first content feature.

15. The method of claim 14, wherein the generating the output feature of the (i+1)-th resolution layer by using the fusion module of the (i+1)-th resolution layer based on the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer comprises:

determining a difference between the first intermediate feature of the (i+1)-th resolution layer and the third content feature under the (i+1)-th resolution layer by using the fusion module of the (i+1)-th resolution layer to obtain a third feature;

fusing the third feature and the first intermediate feature of the (i+1)-th resolution layer by using the fusion module of the (i+1)-th resolution layer to obtain a fourth feature;

obtaining a style mask feature by using the fusion module of the (i+1)-th resolution layer based on the fourth feature and the style degree; and

generating the output feature of the (i+1)-th resolution layer by using the fusion module of the (i+1)-th resolution layer based on the style mask feature, the first intermediate feature of the (i+1)-th resolution layer, and the third content feature under the (i+1)-th resolution layer.

16. The method of claim 15, wherein the determining a total loss of the image processing model based on each generated style image and a style sample corresponding to each style image comprises:

determining an adversarial loss, a first reconstruction loss and a mask feature regularization loss respectively based on each generated style image and the style sample corresponding to each style image, and summing the adversarial loss, the first reconstruction loss and the mask feature regularization loss to obtain the total loss,

the adversarial loss being an adversarial loss of the second generative subnetwork, the first reconstruction loss being a loss of the generated style image relative to the corresponding style sample, and the mask feature regularization loss being used to constrain an average value of elements in the style mask feature.

17. An apparatus for training an image processing model training apparatus comprising a style encoder, a first generative subnetwork and a second generative subnetwork, wherein the style encoder is configured to map a style exemplar or real sample to a style code, the first generative subnetwork comprises a fully convolutional network, and is configured to generate a first content feature of the real sample, and the second generative subnetwork is configured to generate a style image of the real sample based on the style code and the first content

feature,

the apparatus comprising:

a first acquisition module, configured to acquire a style sample set and a real sample set;

a first training module, configured to train the first generative subnetwork based on the real sample set such that the first content feature generated by the first generative subnetwork is capable for reconstruction of a structural feature of the style image; and

a second training module, configured to, after training of the first generative subnetwork completes, train the image processing model iteratively based on the style sample set, the real sample set and the style code until a first convergence condition is satisfied.

18. An image processing apparatus, applied to an image processing model, the image processing model comprising a style encoder, a first generative subnetwork and a second generative subnetwork, wherein the style encoder is configured to map a style exemplar or content image to a style code, the first generative subnetwork comprises a fully convolutional network, and is configured to generate a first content feature of the content image, and the second generative subnetwork is configured to generate a style image of the content image based on the style code and the first content feature, the apparatus comprising:

a first generation module, configured to generate the style code of the content image by using the style encoder, the content image being an image determined from an image frame set to be processed, and the image frame set having image frames with a sequential relationship;

a reconstruction module, configured to generate, based on the content image by using the first generative subnetwork, the first content feature capable for reconstruction of a structural feature of the style image; and

a second generation module, configured to generate the style image by using the second generative subnetwork based on the first content feature and the style code.

19. A computer device, comprising a memory and a processor,

wherein the memory stores a computer program capable of running in the processor;
and

the processor executes the computer program to implement the steps in the method of any one of claims 1-6, or, implement the steps in the method of any one of claims 7-16.

20. A computer storage medium, storing one or more programs capable of being executed by one or more processors to implement the steps in the method of any one of claims 1-6, or, implement the steps in the method of any one of claims 7-16.