

IMAGE CONVERSION METHOD AND APPARATUS, DEVICE, AND STORAGE MEDIUM

TECHNICAL FIELD

[0001] Embodiments of the application relate to the field of image processing, and relate, but not limited, to an image conversion method and apparatus, a device, and a storage medium.

BACKGROUND

[0002] In the related art, image conversion is implemented based on cycle consistency, which requires that after an image is converted from an original domain to a target domain, the image can be converted back to the original domain to implement the reconstruction of the image. However, for conversion between domains with a relatively great form difference, too much information is usually lost after the image conversion, which makes it unlikely to effectively reconstruct an original image and further achieve an ideal effect.

SUMMARY

[0003] The embodiments of the application provide technical solutions to image conversion.

[0004] The technical solutions of the embodiments of the application are implemented as follows.

[0005] The embodiments of the application provide an image conversion method, which may include the following operations.

[0006] A first domain corresponding to a first image is determined using an image

conversion network.

- [0007] A feature mapping relationship between the first image and a preset image in each layer of the image conversion network is determined based on the first domain and a second domain corresponding to the preset image.
- [0008] A picture of the first image is converted from the first domain to the second domain based on the feature mapping relationship to obtain a second image.
- [0009] The embodiments of the application provide an image conversion apparatus, which includes a first determination module, a second determination module, and a first conversion module.
- [0010] The first determination module is configured to determine a first domain corresponding to a first image using an image conversion network.
- [0011] The second determination module is configured to determine a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image.
- [0012] The first conversion module is configured to convert a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.
- [0013] Correspondingly, the embodiments of the application provide a computer storage medium having a computer-executable instruction stored therein. The computer-executable instruction is executed to implement the steps of the method described above.
- [0014] The embodiments of the application provide an electronic device, which includes a memory and a processor. A computer-executable instruction may be stored in the memory. The processor runs the computer-executable instruction in the memory to implement the steps of the method.
- [0015] According to the image conversion method and apparatus, device, and storage medium provided in the embodiments of the application, the domain that the picture of

the acquired first image belongs to is analyzed at first using the image conversion network, and then the same feature class of the first domain and the second domain corresponding to the preset image is analyzed to determine a content mapping relationship between the first image and a second image in each layer, so that the picture of the first image may be converted from the first domain to the second domain based on the content mapping relationship in each layer to obtain a converted second image. As such, the content mapping relationship between the first image and the preset image in each layer may be established to implement natural and accurate image conversion.

BRIEF DESCRIPTION OF THE DRAWINGS

- [0016] FIG. 1 is an implementation flowchart of an image conversion method according to an embodiment of the application.
- [0017] FIG. 2 is another implementation flowchart of an image conversion method according to an embodiment of the application.
- [0018] FIG. 3 is a schematic diagram of an application scene of an image conversion method according to an embodiment of the application.
- [0019] FIG. 4 is an architecture diagram of a network for distilling generative prior according to an embodiment of the application.
- [0020] FIG. 5 is a structure diagram of an image conversion network according to an embodiment of the application.
- [0021] FIG. 6 is a structure composition diagram of an image conversion apparatus according to an embodiment of the application.
- [0022] FIG. 7 is a composition structure diagram of an electronic device according to an embodiment of the application.

DETAILED DESCRIPTION

[0023] In order to make the purposes, technical solutions, and advantages of the embodiments of the application clearer, specific technical solutions of the disclosure will further be described below in combination with the drawings in the embodiments of the application in detail. The following embodiments are adopted to describe the application rather than limit the scope of the application.

[0024] "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 subset or different subsets of all the possible embodiments, and may be combined without conflicts.

[0025] Term "first/second/third" involved in the following descriptions 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 to implement the embodiments of the application described herein in sequences except the illustrated or described ones.

[0026] Unless otherwise defined, all technical and scientific terms used herein have the same meanings as commonly understood by those skilled in the art of the application. Terms used in the application are only adopted to describe the embodiments of the application and not intended to limit the application.

[0027] Nouns and terms involved in the embodiments of the application will be described before the embodiments of the application are further described in detail. The nouns and terms involved in the embodiments of the application are suitable to be explained as follows.

[0028] 1) Conditional Generative Adversarial Net (cGAN): a generator of the cGAN takes a noise and a label as an input, and a discriminator takes a generated object, an original object, and a label as an input. In addition, the cGAN optimizes a determination criterion of the discriminator to determine false data with any label and true data with an error label to be false and determine true data with a correct label to be true.

[0029] 2) Random noise: the random noise is a random variable of a Gaussian distribution. An adversarial network implements the registration transformation of a

distribution. A generator of the adversarial network transforms a Gaussian distribution (or another prior random distribution) into a target data distribution. Here, an input random variable is practically considered as a hidden space for data.

[0030] An exemplary application of an image conversion system provided in the embodiments of the application will be described below. A terminal in the system provided in the embodiments of the application may be various types of electronic devices with an image acquisition function, such as a notebook computer, a tablet computer, a desktop computer, and a mobile device (e.g., a personal digital assistant, a dedicated messaging device, and a portable game device).

[0031] An exemplary application in case of implementing the image conversion system as an electronic device will be described below.

[0032] FIG. 1 is an implementation flowchart of an image conversion method according to an embodiment of the application. As shown in FIG. 1, descriptions will be made in combination with the operations shown in FIG. 1.

[0033] In S101, a first domain corresponding to a first image is determined using an image conversion network.

[0034] In some embodiments, the first image may be an image acquired in any scene; the first image may be an image with a complex picture content, or an image with a simple picture content. The first image is input to the image conversion network to determine a domain that a picture of the first image belongs to. The first domain corresponding to the first image is the domain that a picture content of the first image belongs to. The picture content may refer to at least one of a content represented by an image region of which an area proportion exceeds a preset threshold in the image, a content indicated by a foreground, a content indicated by a central region, or the like. A scene of the picture content is analyzed to determine the domain that the picture content belongs to, namely to obtain the first domain corresponding to the first image. The domain that the picture content belongs to refers to a scene type of the picture content. For example, if the content represented by the image region of which the area proportion exceeds the preset threshold in the first image is a cat, it is determined that the first

domain is cat. Alternatively, if the content picture indicated by the foreground of the first image is a building, it is determined that the first domain is building. The image conversion network is configured to implement the conversion of the picture of the first image from the first domain to a second domain corresponding to a preset image, and is obtained by determining prior information between a first sample image and a second sample image and establishing a mapping relationship between content features in multiple layers based on the content feature in each layer, thereby introducing the mapping relationship between the content features in the multiple layers to a generator for training.

[0035] In S102, a feature mapping relationship between the first image and a preset image in each layer of the image conversion network is determined based on the first domain and a second domain corresponding to the preset image.

[0036] In some embodiments, the preset image is an image of which a picture content belongs to the second domain. The preset image may be an image acquired in the second domain, or a random image generated through a random noise. A picture content of an image indicated by the first domain is different from a picture content of an image indicated by the second domain. In some possible implementation modes, a class of an object in a picture content in an image is analyzed to determine a domain corresponding to the image. For example, if the picture content in the first image is a dog, it is determined that the first domain corresponding to the first image is dog. If the picture content in the preset image is a face, it is determined that the second domain corresponding to the preset image is portrait. The first domain corresponding to the first image and the second domain corresponding to the preset image may be analyzed to determine the same feature classes in the picture of the first image and a picture of a second image and further establish a mapping relationship between these feature classes. Each layer of the image conversion network represents a layer for feature extraction. The feature mapping relationship in each layer is described taking a single layer in the image conversion network as an example. A feature mapping relationship in a layer is configured to represent a mapping relationship between content features of the same class of the first image and the preset image in this layer. For example, the first image is a dog face image, the preset image is a cat face image, and a corresponding relationship

between a dog facial feature class in the first image and a cat facial feature class in the preset image and a corresponding relationship between an orientation of the dog in the first image and an orientation of the cat in the preset image represent mapping relationships between content features of the first image and the preset image.

[0037] In S103, a picture of the first image is converted from the first domain to the second domain based on the feature mapping relationship to obtain a second image.

[0038] In some embodiments, the feature mapping relationship in each layer is input to a generator of the image conversion network to implement the conversion of the first image from the first domain to the second domain to generate the second image.

[0039] In the embodiment of the application, the domain that the picture of the acquired first image belongs to is analyzed at first, and then the same feature class of the first domain and the second domain corresponding to the preset image is analyzed to determine a content mapping relationship between the first image and the second image in each layer, so that the picture of the first image may be converted from the first domain to the second domain based on the content mapping relationship to obtain a converted second image. As such, the content mapping relationship between the first image and the preset image in each layer may be established to implement natural and accurate image conversion.

[0040] In some embodiments, a cross-layer connection is established between a content encoder and generator of the image conversion network, a content feature and style feature in each layer are extracted, and the picture of the first image is converted from the first domain to the second domain in combination with the feature mapping relationship in each layer to obtain the second image. That is, S103 may be implemented through the following S131 and S132 (not shown in the figure).

[0041] In S131, a style feature of the preset image in each layer in the image conversion network is determined using a style encoder in the image conversion network.

[0042] In some possible implementation modes, the preset image is input to the style encoder of the image conversion network to generate the style feature of the preset image in each layer.

- [0043] In S132, the style feature and corresponding feature mapping relationship in any layer in the image conversion network are determined as a feature group.
- [0044] In some possible implementation modes, for a layer in the image conversion network, the style feature in the layer and the feature mapping relationship between the first image and the preset image in the layer are merged as a feature group. As such, a feature group in each layer may be obtained.
- [0045] In S133, a feature group in each layer in the image conversion network is input to the generator of the image conversion network to obtain the second image.
- [0046] In some possible implementation modes, the style feature of the preset image corresponding to the second domain is extracted using the style encoder in the image conversion network. The style feature is configured to represent an image style in the second domain, e.g., a background picture, a color, a tone, a texture, brightness, and a foreground picture. The feature mapping relationship and the style feature in the second domain are input to the generator of the image conversion network to generate the second image. Accordingly, the introduced feature mapping relationship in each layer is combined with the style feature of the preset image, so that the domain conversion of the image may be implemented more accurately.
- [0047] In some embodiments, for establishing the mapping relationship between the content features in each layer, a feature class shared between two domains is analyzed to determine the feature mapping relationship between the first image and the preset image in each layer. That is, S102 may be implemented through the operations shown in FIG. 2. FIG. 2 is another implementation flowchart of an image conversion method according to an embodiment of the application. The following descriptions are made in combination with the operations shown in FIG. 2.
- [0048] In S201, a feature class shared by the first image and the preset image in each layer of the image conversion network is determined based on the first domain and the second domain.
- [0049] In some embodiments, each layer represents a layer for the feature extraction of the image. For example, feature extraction is performed on the first image and the

preset image through the content encoder, and it is determined that a layer of the content encoder is a layer for the feature extraction of the image. An intrinsic connection between the two domains is analyzed, and the same high-layer feature class and intermediate-layer feature class of the first image and the preset image are determined based on the intrinsic connection. For example, the first domain is cat, the second domain is dog, and the intrinsic connection between the two domains is a connection between abstract content feature classes and a connection between concrete content feature classes. The abstract content feature class includes one or more of a face orientation, a position of a picture content in an image, etc. The same concrete content feature class of the first domain and the second domain includes one or more of a facial feature class, a limb feature class, a body feature class, etc.

[0050] In some possible implementation modes, a content feature shared by the first image and the preset image in each layer is determined using the image conversion network. The image conversion network includes a content encoder. The content encoder is used to perform feature extraction on the first domain corresponding to the first image and the preset image in any layer of the content encoder to obtain an image feature of the first image in this layer and an image feature of the preset image in this layer. Based on this, a common content feature class of the first domain corresponding to the first image and the preset image in any layer is determined.

[0051] In some embodiments, an intrinsic connection between the two domains is analyzed to determine a high-layer feature class and intermediate-layer feature class between the first image and the preset image, thereby obtaining the feature class shared in each layer. That is, S201 may be implemented through the following operations S211 to S213 (not shown in the figure).

[0052] In S211, an abstract content feature class shared by the first image and the preset image is determined based on the first domain and the second domain.

[0053] In some embodiments, an intrinsic connection between the first domain and the second domain is analyzed to determine abstract content feature classes, which are semantically the same, of the first image and the preset image, e.g., an orientation and position of the picture in the first image and an orientation and position of the picture in

the preset image.

[0054] In S212, a concrete content feature class shared by the first image and the preset image in each layer is determined.

[0055] In some embodiments, the number of layers of the content encoder for the feature extraction of the first image is determined, so as to determine the same concrete content feature class in content features of an image content of the first image and an image content of the second image in each layer. The concrete content feature class includes a class of a concrete content feature that both the first image and the second image have. For example, if the first image is a face image with a facial feature, and the preset image is a cat image with a facial feature, facial feature is a concrete content feature class shared by the first image and the preset image in an intermediate layer of the image conversion network.

[0056] In S213, the feature class in each layer is obtained based on the abstract content feature class and the concrete content feature class shared in each layer.

[0057] In some embodiments, the abstract content feature class taken as a high-layer content feature class and the concrete content feature class shared in each layer are determined as feature classes shared by the first image and the preset image. Accordingly, the intrinsic connection between the first domain corresponding to the first image and the second domain corresponding to the preset image is analyzed to determine the abstract content feature class shared by the first image and the preset image and the concrete content feature class in each layer. Therefore, the feature class shared by the first image and the preset image in each layer may be obtained accurately.

[0058] After S201, the following operations are executed taking each layer of the image conversion network as a target layer.

[0059] In S202, a first output feature of the first image in the target layer and a second output feature of the preset image in the target layer are determined based on the feature class shared in the target layer respectively.

[0060] In some embodiments, the first output feature of the first image in the target

layer is determined based on the feature class shared in the target layer. For each layer of the image conversion network, feature extraction is performed on the first image in this layer to obtain an encoder feature of the feature class of the layer, and the encoder feature and a generator feature output by the generator are fused to obtain a final output feature in this layer, i.e., the first output feature. Similarly, for each layer of the image conversion network, feature extraction is performed on the preset image in this layer to obtain an encoder feature of the feature class of the layer, and the encoder feature and a generator feature output by the generator are fused to obtain a final output feature in this layer, i.e., the second output feature.

[0061] In some possible implementation modes, a content feature output by the encoder and a generator feature output by the generator are fused through mask information to obtain an output feature finally output by the layer. That is, the operation in S202 that "a first output feature of the first image in the target layer is determined based on the feature class shared in the target layer" may be implemented through the following S221 to S223 (not shown in the figure).

[0062] In S221, an encoder feature of the feature class shared in the target layer in the first image is determined using a content encoder of the image conversion network.

[0063] In some embodiments, the first image is input to the content encoder of the image conversion network to obtain a content feature of the first image corresponding to the same feature class of the first image and the preset image in the target layer of the content encoder, i.e., the encoder feature. The content feature of the first image is extracted using the content encoder in the image conversion network to obtain the encoder feature belonging to the feature class. For example, the first image is a cat face image, the preset image is a dog face image, the feature class in each layer includes a facial feature class, a face orientation, position information, etc., and the encoder feature includes a facial feature, an orientation feature, a position feature, etc.

[0064] In S222, the encoder feature in the target layer is input to a generator of the image conversion network to determine a generator feature in the target layer.

[0065] In some embodiments, the encoder feature in the target layer is input to the

generator of the image conversion network to obtain the generator feature in this layer. In other embodiments, a style feature, extracted by a style encoder of the image conversion network, of the preset image is also input to the generator at the same time of inputting the encoder feature in the target layer to the generator such that the generator generates the generator feature in the target layer.

[0066] In S223, the generator feature in the target layer and the encoder feature in the target layer are fused to obtain the first output feature of the first image in the target layer.

[0067] In some embodiments, in any layer of the image conversion network, a generator feature generated by the generator and an encoder feature output by the content encoder are fused to obtain a final output feature in this layer, i.e., the first output feature. In this manner, an intermediate-layer feature of the first image is introduced to the generator through a cross-layer connection between the content encoder and the generator before a high-layer content feature of the first image is extracted using the content encoder, so that a mapping relationship between the high-layer feature and the intermediate-layer feature may be established. Therefore, features in multiple layers are introduced to the generator to ensure that the generator has more detailed features and the second image generated by the generator has more detailed features to further implement the domain conversion from the first image to the preset image more accurately.

[0068] In other embodiments, an implementation process of the operation in S202 that "a second output feature of the preset image in the target layer is determined based on the feature class shared in the target layer" is similar to that of S221 to S223. That is, a second content feature of the preset image in the target layer is extracted through the content encoder of the image conversion network at first. In some possible implementation modes, if the preset image is a random image generated through a random noise, the second content feature is a content feature, acquired by the content encoder, of the random image. If the preset image is an acquired image with a preset picture, the second content feature is a feature belonging to the feature class in each layer in the preset picture. Then, the second content feature is input to the generator to generate a generator feature in this layer. Finally, the generator feature output by the generator and the content feature output by the content encoder are fused to obtain the second output

feature, output by this layer, of the preset image.

[0069] In S203, the feature mapping relationship between the first image and the preset image in the target layer is determined based on the first output feature and the second output feature.

[0070] In some embodiments, the feature mapping relationship between the content features of the first image and the preset image in each layer is determined in the image conversion network. The feature mapping relationship may represent a corresponding relationship between the content features, belonging to the same feature class, of the first image and the preset image in the target layer. Therefore, the feature class shared in the two domains may be analyzed to accurately establish the mapping relationship between the first image and the preset image in each layer to further convert the first image into the second image corresponding to the second domain accurately.

[0071] In some embodiments, a concrete content feature in an intermediate layer is updated based on a cross-layer connection between the content encoder and the generator through a hidden state of the cross-layer connection to obtain a content feature in this layer. That is, S221 may be implemented through the following operations.

[0072] At a first step, an abstract content feature belonging to an abstract content feature class in the feature class shared in the target layer and a concrete content feature belonging to a concrete content feature class in the feature class shared in the target layer in the first image are determined using the content encoder.

[0073] In some embodiments, an abstract content feature class is determined in the feature class shared in each layer, namely a feature class shared by the first image and the preset image in a high layer of the image conversion network is determined in the feature class shared in each layer. A content feature corresponding to the feature class in the high layer is the abstract content feature. The concrete content feature of the concrete content feature class in the feature class shared in the target layer is a content feature of the first image corresponding to a feature class shared by the first image and the preset image in an intermediate layer or low layer of the image conversion network.

[0074] At a second step, an initial hidden state is determined based on the abstract

content feature.

[0075] In some possible implementation modes, the high-layer content feature, extracted by the content encoder, of the first image, i.e., the abstract content feature, is determined as the initial hidden state in this layer.

[0076] At a third step, the concrete content feature in the target layer is updated based on the initial hidden state to obtain the encoder feature in the target layer.

[0077] In some possible implementation modes, the initial hidden state and a content feature in a second layer are fused to obtain a hidden state in the second layer. That is, a hidden state in an i -th layer may be obtained by inputting a hidden state in an $(i-1)$ th layer and a content feature (i.e., concrete content feature) in the i -th layer to an activation layer of an update network R , convolving the hidden state in the $(i-1)$ th layer and the concrete content feature in the i -th layer, inputting a convolution result to an activation function of the activation layer, and performing element-by-element multiplication on an output of the activation function and the hidden state in the $(i-1)$ th layer. The hidden state in the i -th layer and the concrete content feature in the i -th layer are connected in cascade and input to a feature update network U for a convolution operation, and a convolution result is input to the activation function, thereby updating the concrete content feature in the i -th layer to obtain an encoder feature in the i -th layer to further obtain an encoder feature in each layer. Accordingly, a hidden state in a next layer is updated through a hidden state in a previous layer, and an operation such as convolution is performed on a hidden state and concrete content feature in the same layer to update the concrete content feature to obtain an encoder feature output by the encoder in this layer. Therefore, the content feature of the first image in the intermediate layer of the network is introduced to the network.

[0078] In some embodiments, encoder features of the first image in multiple layers are introduced through a dynamic cross-layer connection established between the content encoder and the generator, so that a final output feature in each layer may be obtained. That is, S222 may be implemented through the following operations.

[0079] At a first step, mask information of the first image in the target layer is

determined based on a hidden state in the target layer and the encoder feature in the target layer.

[0080] In some embodiments, an abstract content feature of a high layer is taken as an initial hidden state of the first image and input together with a concrete content feature in any layer of intermediate layers to a mask network to obtain mask information of this layer. The mask information of this layer is configured to select a useful content feature from content features in this layer, namely the target content feature, usually referring to a content feature in a main region in the image, i.e., a content feature, except redundant information, in the content feature, for input to the generator to generate the second image. The redundant information refers to a content feature corresponding to a content which does not directly point to the domain in the image. For example, when the domain is building, all content features corresponding to non-building objects such as trees and the blue sky in the image may be considered as redundant information.

[0081] In some possible implementation modes, after a dynamic cross-layer connection is established between the content encoder and generator of the image conversion network, a hidden state of the dynamic connection and a content feature in any layer may be acquired to determine mask information of this layer. That is, the first step may be implemented through the following operations.

[0082] In step 1, the abstract content feature is determined as a hidden state in a first layer when the target layer is the first layer, or, a hidden state in an i -th layer is determined based on a hidden state in an $(i-1)$ th layer and a concrete content feature in the i -th layer when the target layer is the i -th layer.

[0083] In some possible implementation modes, i is an integer greater than 1. The hidden state in the first layer is the abstract content feature when i is 2. That is, the initial hidden state is the content feature, extracted in the high layer of the image conversion layer, of the first image. The initial hidden state and a concrete content feature in a second layer are connected in cascade, then input to the activation layer of the image conversion network, and up-sampled and convolved to obtain a hidden state in the second layer. Similarly, the hidden state in the second layer may be updated based on the hidden state in the second layer and a concrete content feature in a third layer to obtain a

hidden state in the third layer. By analogy, a hidden state in any layer may be obtained.

[0084] In step 2, mask information of the first image in a $(j+1)$ th layer is determined based on a hidden state in a j -th layer and a concrete content feature in the $(j+1)$ th layer.

[0085] In some possible implementation modes, j is an integer greater than or equal to 1. The hidden state in the j -th layer and the concrete content feature in the $(j+1)$ th layer are connected in cascade, then input to an activation layer of the mask network, and up-sampled and convolved to obtain the mask information in the $(j+1)$ th layer. In some possible implementation modes, for selecting a valid portion from the content feature through the obtained mask information more accurately, regularization processing is performed on the mask information to make the mask information as sparse as possible. Accordingly, the mask information of the first image may be determined through the hidden state and content feature in any layer in the dynamic cross-layer connection, which is favorable for selecting the target content feature from the content feature of the first image to increase the image conversion speed of the image conversion network.

[0086] At a second step, the concrete content feature in the target layer and the mask information in the target layer are input to the generator to generate the generator feature in the target layer.

[0087] In some embodiments, a valid portion in the intermediate-layer content feature is selected through the mask information in the target layer and input to the generator. A feature corresponding to numerical value 1 is selected from a concrete content feature in any layer through a value in mask information in this layer, and is input to the generator to obtain a generator feature of the generator in this layer.

[0088] In some possible implementation modes, for reducing the redundant information input to the generator and calculations of the generator, the second step may be implemented through the following operations.

[0089] In step a, regularization processing is performed on the mask information in the target layer to obtain sparse mask information in the target layer.

[0090] In some possible implementation modes, regularization processing is

performed on mask information in any layer using Layer 2 (L2) or Layer 1 (L1) regularization to make the mask information as sparse as possible to obtain sparse mask information in this layer. Since a vector corresponding to a feature such as a background in the mask information changes to 0 in a process of making the mask information sparse to obtain the sparse mask information, namely the redundant information of the picture in the image in the mask information is discarded, the sparse mask information includes the target content feature in the target layer, and furthermore, invalid information in the mask information is reduced. For example, if the first image is a cat face image, information, except a facial feature of the cat, in the image, i.e., background information, is invalid information.

[0091] In step b, a target content feature matched with the sparse mask information in the target layer is acquired from the concrete content feature in the target layer.

[0092] In some possible implementation modes, the target content feature matched with the sparse mask information in the target layer is selected from the concrete content feature in the target layer. The target content feature matched with the sparse mask information in the target layer may be understood as selecting a content feature corresponding to value 1 in the sparse mask information from the content feature in this layer using the sparse mask information. Therefore, the selected feature includes very little redundant information, and the content features in multiple layers may be screened automatically.

[0093] In step c, the target content feature in the target layer is input to the generator to generate the generator feature in the target layer.

[0094] In some possible implementation modes, the target content feature in this layer is input to the generator for the generator to output the generator feature in this layer. Accordingly, a valid portion is selected from the concrete content feature in this layer using the sparse mask information as an input of the generator, so that calculations of the generator are reduced.

[0095] At step 3, the encoder feature in the target layer and the generator feature in the target layer are fused based on the mask information in the target layer to obtain the first

output feature of the first image in the target layer.

[0096] In some embodiments, in any layer, an encoder feature and a generator feature are fused using mask information in this layer to obtain a final output feature in this layer, i.e., the first output feature. As such, a dynamically screened valid feature in each layer is introduced between the content encoder and the generator.

[0097] In some possible implementation modes, the operation that the encoder feature and the generator feature are fused using the mask information and adversarial information of the mask information, i.e., step 3, may be implemented through the following process.

[0098] First, the encoder feature in the target layer is adjusted based on the mask information in the target layer to obtain a first adjusted feature in the target layer.

[0099] In some possible implementation modes, in any layer, element-by-element multiplication is performed on mask information and encoder feature in this layer to obtain a first adjusted feature in this layer.

[00100] Then, the generator feature in the target layer is adjusted based on adversarial information of the mask information in the target layer to obtain a second adjusted feature in the target layer.

[00101] In some possible implementation modes, the adversarial information of the mask information in the target layer may be a sum of an opposite number of the mask information and a preset value (for example, the preset value is 1). Element-by-element multiplication is performed on the adversarial information and the generator feature to obtain the second adjusted feature in this layer.

[00102] Finally, the first adjusted feature in the target layer and the second adjusted feature in the target layer are fused to obtain the first output feature in the target layer.

[00103] In some possible implementation modes, in any layer, element-by-element summation is performed on the obtained first adjusted feature and second adjusted feature to obtain a final first output feature in this layer. Accordingly, the encoder feature

and generator feature in the target layer may be fused through the mask information, so that the introduction of the content feature in the intermediate layer of the network may be implemented.

[00104] In some embodiments, the image conversion network is obtained by training a network to be trained based on a first sample image of which a picture belongs to a first sample domain and a second sample image of which a picture belongs to a second sample domain. A training process of the network to be trained may be implemented through the following S151 to S156 (not shown in the figure).

[00105] In S151, the first sample image and second sample image which are correlated with a picture are acquired.

[00106] In some embodiments, there is correlated prior information between the first sample image and the second sample image and a certain correlation between the picture of the first sample image and the picture of the second sample image. For example, the first sample image and the second sample image are generated based on the same random noise. Orientations, poses, and other information of objects in the pictures of the images are the same.

[00107] In some possible implementation modes, the first sample image and the second sample image may be acquired through the following process.

[00108] First, a preset noise and a domain label set at least including the first sample domain and the second sample domain are acquired.

[00109] In some possible implementation modes, the preset noise may be a Gaussian random noise. As shown in FIG. 3, the preset noise may be Z_1 , Z_2 , or Z_3 . The domain label set includes multiple domain labels. As shown in FIG. 3, a bird domain, a cat domain, a dog domain, and the like are included.

[00110] Finally, the preset noise and the domain label set are input to a trained adversarial network to generate the first sample image and second sample image which have matched semantic information.

[00111] In some possible implementation modes, the trained adversarial network may be a big Generative Adversarial Network (bigGAN). The preset noise and the domain label set are input to the adversarial network as conditional information, and sample images corresponding to different domain labels under each random noise are output to further obtain the first sample image and the second sample image. Accordingly, sample images are generated using a cGAN based on the same random noise and two random domains, so that the trained content encoder may extract content features shared by the sample images to facilitate the reconstruction of shapes and appearances of the sample images.

[00112] In S152, a first sample feature of the first sample image in each layer of a trained content encoder and a second sample feature of the second sample image in each layer are determined using the trained content encoder in the network to be trained.

[00113] In some embodiments, the trained content encoder in the network to be trained is obtained by training based on the first sample image and second sample image which have prior information. The first sample image and the second sample image are input to two trained content encoders respectively to obtain a content feature of the first sample image in each layer of the content encoder, i.e., the first sample feature, and a content feature of the second sample image in each layer of the content encoder, i.e., the second sample feature.

[00114] In some possible implementation modes, a content encoder to be trained is trained based on the first sample image and the second sample image to obtain the trained content encoder, which may be implemented through the following operations.

[00115] At a first step, a first predicted content feature of the first sample image and a second predicted content feature of the second sample image are determined using the content encoder to be trained.

[00116] In some possible implementation modes, the first sample image and the second sample image are input to the content encoder to be trained. An overall content feature of the first sample image, i.e., the first predicted content feature, is extracted using the content encoder to be trained. Similarly, an overall content feature of the second sample

image, i.e., the second predicted content feature, is extracted using the content encoder to be trained.

[00117] At a second step, the first predicted content feature, a sample style feature of the first sample image, and the first sample domain are input to a decoder to reconstruct the first sample image to obtain a second reconstructed sample image.

[00118] In some possible implementation modes, the first sample image is input to a style encoder to extract the sample style feature of the first sample image. The first predicted content feature and sample style feature of the first sample image and a domain label of the first sample domain are input to the decoder for the decoder to reconstruct the first sample image to obtain the second reconstructed image.

[00119] At a third step, a first loss between the second reconstructed sample image and the first sample image is determined.

[00120] In some possible implementation modes, a difference between a picture content of the second reconstructed sample image and a picture content of the first sample image is analyzed to determine the first loss. That is, the first loss is used to measure a similarity between the second reconstructed image and the first sample image.

[00121] At a fourth step, a second loss between the second predicted content feature and the first predicted content feature is determined.

[00122] In some possible implementation modes, a difference between the second predicted content feature of the second sample image and the first predicted content feature of the first sample image is analyzed to obtain the second loss. That is, the second loss is used to represent a similarity between the second sample image and the first sample image.

[00123] At a fifth step, a network parameter of the content encoder to be trained is adjusted based on the first loss and the second loss to obtain the trained content encoder.

[00124] In some possible implementation modes, the first loss and the second loss are combined to adjust the network parameter of the content encoder to be trained such that

the trained content encoder may extract a high-layer content feature shared by the first sample image and the second sample image. Accordingly, the content encoder to be trained is trained using a sample image set with prior information, so that the trained content encoder may extract the same high-layer content feature more accurately.

[00125] In S153, a sample feature mapping relationship between the first sample image and the second sample image in each layer of the trained content encoder is determined based on the first sample feature in each layer and the second sample feature in each layer.

[00126] In some embodiments, the first sample feature of the first sample image in each layer and the second sample feature of the second sample image in each layer are extracted through the trained encoder, and a corresponding relationship between the first sample feature and the second sample feature is determined, thereby obtaining sample feature mapping relationships in multiple layers.

[00127] In S154, the sample feature mapping relationship in each layer and a sample style feature of the second sample image are input to a first generator of the network to be trained to obtain a converted sample image after the picture of the first sample image is converted from the first sample domain to the second sample domain.

[00128] In some embodiments, in any layer, the sample feature mapping relationship in this layer and the sample style feature of the second sample image are input to the first generator, thereby predicting a converted sample image after the picture of the first sample image is converted from the first sample domain to the second sample domain.

[00129] In S155, the sample style feature of the second sample image and the second sample feature in each layer are input to a second generator of the network to be trained to obtain a first reconstructed sample image after the picture of the second sample image is reconstructed.

[00130] In some embodiments, the sample style feature of the second sample image and the second sample feature in each layer are input to the second generator to predict the first reconstructed sample image after the second sample image is reconstructed.

[00131] In S156, a network parameter of the network to be trained is adjusted based on a loss of the converted sample image and a loss of the first reconstructed sample image to obtain the image conversion network.

[00132] In some embodiments, the loss of the converted sample image is determined based on a difference between pictures of the converted sample image and the second sample image. The loss of the first reconstructed sample image is determined based on a difference between pictures of the first reconstructed sample image and the second sample image. Based on this, the network parameter such as a weight of the network to be trained is adjusted to obtain the image conversion network.

[00133] An implementation mode of training the network to be trained to obtain the image conversion network is provided through S151 to S156. In the mode, sample images with a prior correlation are obtained to obtain the trained content encoder by training at first, and then a cross-layer connection is established between the content encoder and the generator, thereby introducing content features of the sample images in an intermediate layer to further improve the accuracy of the image conversion network obtained by training.

[00134] In some possible implementation modes, a difference between the converted sample image and the first sample image, a difference between the first reconstructed sample image and the second sample image, and an adversarial loss of the whole network are determined respectively to implement the adjustment of the network parameter of the network to be trained. That is, S156 may be implemented through the following operations.

[00135] At a first step, the loss of the converted sample image is determined based on a content feature of the converted sample image and a content feature of the first sample image.

[00136] In some embodiments, a difference between an overall content feature of the converted sample image and overall content feature of the first sample image, which are extracted by the trained content encoder, is determined at first. Then, a network parameter of the first generator in the network to be trained is adjusted taking the

difference as the loss of the converted sample image to make the loss of the converted sample image generated by the first generator satisfy a convergence condition, thereby obtaining a trained first generator.

[00137] At a second step, the loss of the first reconstructed sample image is determined based on a style feature of the first reconstructed sample image and the style feature of the second sample image.

[00138] In some embodiments, the style feature of the first reconstructed sample image and the style feature of the second sample image are determined using a style encoder to be trained respectively. The first reconstructed sample image is input to the style encoder to extract the style feature of the first reconstructed sample image. Similarly, the style feature of the second sample image is extracted using the style encoder. A difference between the two style features is determined as the loss of the first reconstructed sample image, and network parameters of the second generator and the style encoder are adjusted using the loss of the first reconstructed sample image such that a trained style encoder and second generator may generate reconstructed sample images which cannot be determined to be true or false by a discriminator.

[00139] At a third step, an adversarial loss between the first reconstructed sample image and the second sample image is determined.

[00140] In some embodiments, the first reconstructed sample image and the second sample image are input to the discriminator of the network to be trained respectively to obtain an output of the first reconstructed sample image and an output of the second sample image. Then, opposite numbers of the output of the second sample image and the output of the first reconstructed sample image are fused to obtain the adversarial loss such that the first reconstructed sample image and the second sample image are indistinguishable for the discriminator.

[00141] At a fourth step, the network parameter of the network to be trained is adjusted based on the loss of the converted sample image, the loss of the first reconstructed sample image, and the adversarial loss to obtain the image conversion network.

[00142] In some possible implementation modes, the image conversion network is

obtained by adjusting the network parameter of the first generator based on the loss of the converted sample image, adjusting the network parameter of the style encoder of the network to be trained based on the loss of the first reconstructed sample image, and adjusting the network parameter of the discriminator of the network to be trained based on the adversarial loss. As such, the discriminator, generator, and style encoder in the network to be trained may be trained respectively to obtain the image conversion network with higher image conversion accuracy.

[00143] An exemplary application of the embodiment of the application in a practical application scene will be described below taking the implementation of image conversion of different domains based on a generative prior as an example.

[00144] Unsupervised image conversion is for a purpose of converting an image of a domain into an image of another domain without paired image supervision, and is applied to image edition, image artistry, virtual image realistic rendering, etc. In the related art, a cycle-consistency-based solution is adopted, which requires that an image may be converted back to an original domain after converted from the original domain to a target domain to implement the reconstruction of the image. However, the solution is only suitable for the condition that a similarity between an original domain and a target domain is high, e.g., conversion between a horse and a zebra. For general conversion between domains with relatively great form differences in practical applications, e.g., conversion between any two of a human face, an animal face, and a cartoon face, there is usually lost too much information after image conversion, which makes it unlikely to effectively reconstruct an original image.

[00145] Based on this, the embodiment of the application provides an image conversion method. For the problem that conversion between domains with a relatively great form difference cannot be implemented by a cycle-consistency-based image conversion method, a novel generative-prior-based image conversion method is proposed to establish a robust mapping relationship between different domains to implement conversion between domains with relatively great form differences.

[00146] In the embodiment of the application, a content encoder is trained using prior information of intrinsic connections between different domains provided by a pre-trained

cGAN to extract high-layer content features shared between the domains and further learn intermediate-layer content features in specific tasks, thereby establishing a content mapping relationship in each layer to implement natural and accurate image conversion.

[00147] The image conversion method provided in the embodiment of the application may be implemented through the following two steps.

[00148] At a first step, distillation is performed using the pre-trained cGAN to generate prior information.

[00149] In some embodiments, step 1 may be implemented through the following operations.

[00150] In step 1.1, training data is generated.

[00151] In some possible implementation modes, the training data includes a first sample image and a second sample image. First, a pre-trained cGAN model obtained based on a large-scale dataset is selected. That is, a random noise z and a domain label l are mapped into a generative network of an image of a corresponding domain. For the ease of description, the disclosure takes a pre-trained BigGAN obtained based on dataset ImageNet as an example, but is not limited to the dataset and the network model, which are replaceable components. Then, different noises $z_1, \dots, z_n$ are sampled to generate n images of corresponding domains for domain labels in dataset ImageNet. Totally $k \times n$ images are generated for k domains for subsequent training. As shown in FIG. 3, for dog, cat and bird domains, they are different in form, but images generated based on the same noise (z_1 , z_2 , or z_3) are in highly unified semantic corresponding relationships. For example, images generated based on the same noise z_1 , i.e., an image 301 corresponding to the dog domain, an image 302 corresponding to the cat domain, and an image 303 corresponding to the bird domain, are unified in orientation, pose, etc. These rules provide valid generative priors that may be used to establish relationships between different domains.

[00152] In step 1.2, a content encoder is trained based on the generated training data.

[00153] The content encoder E_c is trained to extract high-layer content features shared by the images generated based on the same z . As shown in FIG. 4, FIG. 4 is an architecture diagram of a network for distilling generative prior according to an embodiment of the application. It can be seen in combination with FIG. 4 that the network includes a content encoder E_c 401 and a decoder F 403. In training, two training images are randomly sampled, including an image $(x = cGAN(z, lx))$ 41 and an image $(y = cGAN(z, ly))$ 42. That is, the image 41 and the image 42 are images generated according to the same noise. Herein, cGAN is the conditional generative adversarial network in the step 1.1, z is the shared noise, and lx and ly are two domain labels respectively. The content encoder E_c 401 performs content feature extraction to obtain a high-layer content feature 43 of the image 41. A content encoder E_c 402 performs content feature extraction on the image 42 to obtain a high-layer content feature 44 of the image 42. A style encoder E_s 403 extracts a style feature sx of the image 42. The decoder F 404 reconstructs the image 41 according to the high-layer content feature 43, the style feature sx , and the domain label lx of the image 41 to obtain an image 45. In the embodiment of the application, information of the style feature sx and the domain label lx is input to F by, but not limited to, adaptive instance normalization to obtain a reconstructed image $F(cx, sx, lx)$. For example, conditional information such as the style feature sx and the domain label lx may also be input to F by path tandem connection.

[00154] It is expected in a training process that the reconstructed sample image may maximally restore the image x and, meanwhile, the content feature extracted from the image y is maximally approximate to that of the training image x , corresponding to L_{rec} and L_{feat} in the following formulas respectively, as shown in formulas (1) and (2):

$$L_{rec} = E_x \left[\| F(cx, sx, lx) - x \|_2^2 \right] \quad (1), \text{ and}$$

$$L_{feat} = E_{(x,y)} \left[\| Ec(y) - Ec(x) \|_1 \right] \quad (2);$$

where E_x represents solving a correlation function depending on variable x . $E_{(x,y)}$

represents solving a correlation function depending on variable (x, y) . $Ec(y)$ and $Ec(x)$ represent the content features, extracted by the content encoder, of the image y and the image x respectively. L_{rec} may also determine a similarity between the images in a measurement manner such as a perceptual loss and a Learned Perceptual Image Path Similarity (LPIPS).

[00155] In some embodiments, besides the reconstructed sample image, L_{rec} may also segment a foreground and background of the training image using a segmentation technology to obtain a foreground mask of the image x , and the foreground mask is reconstructed using the decoder F .

[00156] In the embodiment of the application, an L2 regularization constraint may further be added to the content feature to make the content feature as sparse as possible.

[00157] At a second step, an image is converted based on the generated prior information.

[00158] In some embodiments, a content encoder E_c is fixed, and an image conversion network is trained to implement image conversion. As shown in FIG. 5 which is a structure diagram of an image conversion network according to an embodiment of the application, the image conversion network includes a style encoder E_s 505, a generator 51, a generator 54, a discriminator D 53, and the content encoder E_c 503 and content encoder E_c 504 trained at step 1. The image 501 belongs to domain X . The image 502 belongs to domain Y . The content encoder E_c 503 is used to extract a content feature of the image 501. The content encoder E_c 504 is used to extract a content feature of the image 502. The style encoder E_s 505 is used to extract a style feature of the image 502. A dynamic cross-layer connection is detected between the content encoder E_c 503 and the generator 51, intermediate features, extracted by the content encoder E_c 503, of the image 501 in multiple layers are input to the generator 51, and meanwhile, the style feature of the image 502 is also input to the generator 51, to generate an image 506 after the image 501 is converted from domain X to domain Y . A content encoder E_c 52 is used to judge whether high-layer content features of the image 506 and the image 501 are consistent. The discriminator D 53 is used to determine whether the image 506 and the image 502

are true or false. The content feature of the image 502 and the style feature of the image 502 are input to the generator 54 to implement the reconstruction of the image 502 to obtain a reconstructed sample image 507. Accordingly, the dynamic cross-layer connection is established between the content encoder E_c and the generator G to extract a content feature in each layer, thereby establishing a content mapping relationship between images corresponding to different domains in each layer.

[00159] The second step may be implemented through the following operations.

[00160] In step 2.1, a dynamic cross-layer connection is established between the content encoder and a generator.

[00161] In some possible implementation modes, a valid portion of a content feature in an intermediate layer of E_c is extracted through the dynamic cross-layer connection, and the extracted valid portion and a high-layer content feature extracted by the content encoder E_c are input to the generator G to establish a content mapping relationship in each layer. A mask m is estimated in each layer of the content encoder E_c through the dynamic cross-layer connection, and weighted combination is performed on an encoder feature f_E of the content encoder and a generator feature f_G of the generator through the mask m to implement the introduction of intermediate-layer information.

[00162] The dynamic cross-layer connection includes a hidden state h , an update network R , a mask network M , and a feature update network U . For the ease of description, superscript i is used to represent a generator feature and network corresponding to an i -th layer of the generator G . For an operation in each layer, all features are adjusted to a unified space size at first in a nearest neighbor interpolation manner. For the i -th layer, a hidden state h^i is updated at first using an update network R^i corresponding to the i -th layer, as shown in formula (3):

$$h^i = R^i(h^{i-1}, f_E^i)h^{i-1} \quad (3);$$

where f_E^i represents an encoder feature in the i -th layer. An initial hidden state is set as a high-layer content feature $h^0 = Ec(x)$. Then, mask information m^i in the i -th layer is

determined, as shown in formula (4):

$$m^i = M^i(h^{i-1}, f_E^i) \quad (4).$$

[00163] Next, the encoder feature f_E^i is updated, as shown in formula (5):

$$\hat{f}_E^i = U^i(h^i, f_E^i) \quad (5).$$

[00164] Finally, the encoder feature and a generator feature are fused to obtain a final output f^i of the generator G in the i-th layer, as shown in formula (6):

$$f^i = (1 - m^i) f_G + m^i \hat{f}_E^i \quad (6).$$

[00165] In step 2.2, an image conversion network is trained based on intermediate-layer information introduced through the dynamic cross-layer connection.

[00166] In some possible implementation modes, for a specific task of converting the image from domain X to domain Y, in a training process, the content encoder E_c is fixed, and images $x \in X, y \in Y$ are randomly sampled. The content encoder E_c extracts content features cx and cy from the image x and the image y respectively. E_s extracts style features sx and sy from the image x and the image y respectively. The generator G generates a converted image $\hat{y} = G(cx, sy)$ obtained by converting the image x to domain Y according to the content feature cx of the image x and the style feature sy of the image y .

[00167] A training requirement on the image conversion network is that a content feature of image $\hat{y}$ is consistent with the content feature cs of the image x and a style feature is consistent with the style feature of the image y such that a discriminator network cannot determine whether the converted image $\hat{y}$ is true or false, corresponding to L_{con} , L_{sty} , and L_{adv} in the following formulas respectively, as shown in formulas (7), (8), and (9):

$$L_{con} = E_{x,y} [\| Ec(\hat{y}) - Ec(x) \|_2^2] \quad (7);$$

$$L_{sty} = E_{x,y} [\| F_D(\hat{y}) - F_D(y) \|_1] \quad (8); \text{ and}$$

$$L_{adv} = E_y [\log D(y)] + E_{x,y} [\log(1 - D(\hat{y}))] \quad (9).$$

where F_D is a style feature extracted in an intermediate layer of the network of the discriminator, and is a vector formed by a mean value of each path of the feature in this layer. That is, $F_D(\hat{y})$ and $F_D(y)$ are style features, extracted by the intermediate layer of the network of the discriminator D, of the converted image $\hat{y}$ and the image y .

[00168] In some embodiments, L_{sty} may adopt another loss function for measuring the style consistency, e.g., a Visual Geometry Group (VGG)-network-based style loss and contextual loss in a perceptual loss.

[00169] In some embodiments, $\bar{y} = G(sx, sy)$ reconstructed by the generator G according to the content feature cx and cy of the image y is required to be consistent with the original image y . The consistency of the image y and the reconstructed sample image $\bar{y}$ may be measured through L_{rec} . Alternatively, a similarity between the images is measured by measurement means such as a perceptual loss and an LPIPS. Specifically, L_{rec} is:

$$L_{rec} = E_x [\| \bar{y} - y \|_2^2] \quad (10).$$

[00170] In some embodiments, an L1 regularization constraint may be added to the mask information m to make the mask information as sparse as possible, as shown in the following formula (11):

$$L_{msk} = \sum_i E_x [\| m^i \|_1] \quad (11).$$

[00171] In the embodiment of the application, the distillation network for generating

the prior information may provide highly abstract mapping relationships between images of different domains, so that a strong technical support is provided for conversion between domains with relatively great form differences. Robust mapping relationships between different domains in each layer may be established by content feature extraction in each layer. Therefore, the whole framework may implement image conversion between domains with relatively great form differences, and meanwhile, the generalization to domains except domains providing priors provided by the generative network is supported. Therefore, through the image conversion method provided in the embodiment of the application, artistic face swapping/portrait generation may be implemented. For example, a mapping relationship between a portrait and a cartoon/animal face is established through the technology such that various portraits may be generated for a user, or a real human face may be replaced with the animal face. Intelligent image edition may also be implemented: a specified object in an image is converted into another object. For example, a user in fear of sharp-mouth animals may replace an unwanted bird in a photo with a pet dog/cat without any influence on the layout of the photo. Data augmentation is implemented. For example, a human face dataset is large-scale and comprehensive and includes rich labels, e.g., positioning points, angles and light and shadow of human faces. Through the image conversion method provided in the embodiment of the application, a human face may also be converted into an animal face, and a common label, e.g., the angle and the light and shadow, is reused to implement data augmentation and provide labeled animal face data for deep learning and analysis.

[00172] An embodiment of the application provides an image conversion apparatus. FIG. 6 is a structure composition diagram of an image conversion apparatus according to an embodiment of the application. As shown in FIG. 6, the image conversion apparatus 600 includes a first determination module 601, a second determination module 602, and a first conversion module 603.

[00173] The first determination module 601 is configured to determine a first domain corresponding to a first image using an image conversion network.

[00174] The second determination module 602 is configured to determine a feature

mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image.

[00175] The first conversion module 603 is configured to convert a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.

[00176] In some embodiments, the second determination module 602 includes a first determination submodule, a second determination submodule, and a third determination submodule.

[00177] The first determination submodule is configured to determine a feature class shared by the first image and the preset image in each layer of the image conversion network based on the first domain and the second domain.

[00178] The second determination submodule is configured to, taking each layer as a target layer, determine a first output feature of the first image in the target layer and a second output feature of the preset image in the target layer based on the feature class shared in the target layer respectively.

[00179] The third determination submodule is configured to determine the feature mapping relationship between the first image and the preset image in the target layer based on the first output feature and the second output feature.

[00180] In some embodiments, the first determination submodule includes a first determination unit, a second determination unit, and a third determination unit.

[00181] The first determination unit is configured to determine an abstract content feature class shared by the first image and the preset image based on the first domain and the second domain.

[00182] The second determination unit is configured to determine a concrete content feature class shared by the first image and the preset image in each layer.

[00183] The third determination unit is configured to obtain the feature class in each

layer based on the abstract content feature class and the concrete content feature class shared in each layer.

[00184] In some embodiments, the second determination submodule includes a fourth determination unit, a fifth determination unit, and a first fusion unit.

[00185] The fourth determination unit is configured to determine an encoder feature of the feature class shared in the target layer in the first image using a content encoder of the image conversion network.

[00186] The fifth determination unit is configured to input the encoder feature in the target layer to a generator of the image conversion network to determine a generator feature in the target layer.

[00187] The first fusion unit is configured to fuse the generator feature in the target layer and the encoder feature in the target layer to obtain the first output feature of the first image in the target layer.

[00188] In some embodiments, the fourth determination unit includes a first determination subunit, a second determination subunit, and a first updating subunit.

[00189] The first determination subunit is configured to determine an abstract content feature belonging to an abstract content feature class in the feature class shared in the target layer and a concrete content feature belonging to a concrete content feature class in the feature class shared in the target layer in the first image are determined using the content encoder.

[00190] The second determination subunit is configured to determine an initial hidden state based on the abstract content feature.

[00191] The first updating subunit is configured to update the concrete content feature in the target layer based on the initial hidden state to obtain the encoder feature in the target layer.

[00192] In some embodiments, the fifth determination unit includes a third determination subunit and a first generation subunit.

- [00193] The third determination subunit is configured to determine mask information of the first image in the target layer based on a hidden state in the target layer and the encoder feature in the target layer.
- [00194] The first generation subunit is configured to input the concrete content feature in the target layer and the mask information in the target layer to the generator to generate the generator feature in the target layer.
- [00195] The first fusion unit includes a first fusion subunit.
- [00196] The first fusion subunit is configured to fuse the encoder feature in the target layer and the generator feature in the target layer based on the mask information in the target layer to obtain the first output feature of the first image in the target layer.
- [00197] In some embodiments, the third determination subunit is further configured to: determine the abstract content feature as a hidden state in a first layer when the target layer is the first layer, or, determine a hidden state in an i -th layer based on a hidden state in an $(i-1)$ th layer and a concrete content feature in the i -th layer when the target layer is the i -th layer, where i is an integer greater than 1; and determine mask information of the first image in a $(j+1)$ th layer based on a hidden state in a j -th layer and a concrete content feature in the $(j+1)$ th layer, where j is an integer greater than or equal to 1.
- [00198] In some embodiments, the first generation subunit is further configured to perform regularization processing on the mask information in the target layer to obtain sparse mask information in the target layer, acquire a target content feature matched with the sparse mask information in the target layer from the concrete content feature in the target layer, and input the target content feature in the target layer to the generator to generate the generator feature in the target layer.
- [00199] In some embodiments, the first fusion subunit is further configured to adjust the encoder feature in the target layer based on the mask information in the target layer to obtain a first adjusted feature in the target layer, adjust the generator feature in the target layer based on adversarial information of the mask information in the target layer to obtain a second adjusted feature in the target layer, and fuse the first adjusted feature in the target layer and the second adjusted feature in the target layer to obtain the first

output feature in the target layer.

[00200] In some embodiments, the first conversion module 603 further includes a fourth determination submodule, a first merging submodule, and a first input submodule.

[00201] The fourth determination submodule is configured to determine a style feature of the preset image in each layer in the image conversion network is determined using a style encoder in the image conversion network.

[00202] The first merging submodule is configured to determine the style feature and corresponding feature mapping relationship in any layer in the image conversion network as a feature group.

[00203] The first input submodule is configured to input a feature group in each layer in the image conversion network to the generator of the image conversion network to obtain the second image.

[00204] In some embodiments, the image conversion network is obtained by training a network to be trained based on a first sample image of which a picture belongs to a first sample domain and a second sample image of which a picture belongs to a second sample domain. The apparatus further includes a training module, configured to train the network to be trained to obtain the image conversion network. The training module includes a first acquisition submodule, a fifth determination submodule, a sixth determination submodule, a second input submodule, a third input submodule, and a first adjustment submodule.

[00205] The first acquisition submodule is configured to acquire the first sample image and second sample image of which the pictures are correlated.

[00206] The fifth determination submodule is configured to determine a first sample feature of the first sample image in each layer of a trained content encoder and a second sample feature of the second sample image in each layer using the trained content encoder in the network to be trained.

[00207] The sixth determination submodule is configured to determine a sample feature

mapping relationship between the first sample image and the second sample image in each layer of the trained content encoder based on the first sample feature in each layer and the second sample feature in each layer.

[00208] The second input submodule is configured to input the sample feature mapping relationship in each layer and a sample style feature of the second sample image to a first generator of the network to be trained to obtain a converted sample image after the picture is converted from the first sample domain to the second sample domain.

[00209] The third input submodule is configured to input the sample style feature of the second sample image and the second sample feature in each layer to a second generator of the network to be trained to obtain a first reconstructed sample image after the picture of the second sample image is reconstructed.

[00210] The first adjustment submodule is configured to adjust a network parameter of the network to be trained based on a loss of the converted sample image and a loss of the first reconstructed sample image to obtain the image conversion network.

[00211] In some embodiments, the first acquisition submodule includes a first acquisition unit and a first generation unit.

[00212] The first acquisition unit is configured to acquire a preset noise and a domain label set at least including the first sample domain and the second sample domain.

[00213] The first generation unit is configured to input the preset noise and the domain label set to a trained adversarial network to generate the first sample image and second sample image which have matched semantic information.

[00214] In some embodiments, the apparatus further includes a second training module.

[00215] The second training module is configured to train a content encoder to be trained based on the first sample image and the second sample image to obtain the trained content encoder.

[00216] In some embodiments, the second training module may include a seventh determination submodule, a first reconstruction submodule, an eighth determination

submodule, a ninth determination submodule, and a second adjustment submodule.

[00217] The seventh determination submodule is configured to determine a first predicted content feature of the first sample image and a second predicted content feature of the second sample image using the content encoder to be trained.

[00218] The first reconstruction submodule is configured to input the first predicted content feature, a sample style feature of the first sample image, and the first sample domain to a decoder to reconstruct the first sample image to obtain a second reconstructed sample image.

[00219] The eighth determination submodule is configured to determine a first loss between the second reconstructed sample image and the first sample image.

[00220] The ninth determination submodule is configured to determine a second loss between the second predicted content feature and the first predicted content feature.

[00221] The second adjustment submodule is configured to adjust a network parameter of the content encoder to be trained based on the first loss and the second loss to obtain the trained content encoder.

[00222] In some embodiments, the first adjustment submodule includes a sixth determination unit, a seventh determination unit, an eighth determination unit, and a first adjustment unit.

[00223] The sixth determination unit is configured to determine the loss of the converted sample image based on a content feature of the converted sample image and a content feature of the first sample image.

[00224] The seventh determination unit is configured to determine the loss of the first reconstructed sample image based on a style feature of the first reconstructed sample image and the style feature of the second sample image.

[00225] The eighth determination unit is configured to determine an adversarial loss between the first reconstructed sample image and the second sample image.

[00226] The first adjustment unit is configured to adjust the network parameter of the network to be trained based on the loss of the converted sample image, the loss of the first reconstructed sample image, and the adversarial loss to obtain the image conversion network.

[00227] It is to be noted that 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. Technical details undisclosed in the apparatus embodiment of the application may be understood with reference to the descriptions about the method embodiment of the application.

[00228] It is to be noted that, in the embodiments of the application, the image conversion method may also be stored in a computer-readable storage medium when 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 embodiments of the application substantially or parts making contributions to the conventional 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 configured to enable an electronic device (which may be a terminal, a server, etc.) to execute all or part of the method in each embodiment of the application. 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 application are not limited to any specific hardware and software combination.

[00229] Correspondingly, an embodiment of the application also provides a computer program product, including a computer-executable instruction which may be executed to implement the steps in the image conversion method provided in the embodiments of the application.

[00230] Correspondingly, an embodiment of the application also provides a computer storage medium having stored therein a computer-executable instruction which is executed by a processor to implement the steps of the image conversion method provided in the abovementioned embodiment.

[00231] Correspondingly, an embodiment of the application provides an electronic device. FIG. 7 is a composition structure diagram of an electronic device according to an embodiment of the application. As shown in FIG. 7, the electronic device 700 includes a processor 701, at least one communication bus, a communication interface 702, at least one external communication interface, and a memory 703. The communication interface 702 is configured to implement connections and communications between these components. The communication interface 702 may include a display screen. The external communication interface may include a standard wired interface and wireless interface. The processor 701 is configured to execute an image processing program in the memory to implement the steps of the image conversion method provided in the abovementioned embodiment.

[00232] The above descriptions about the embodiments of the image conversion apparatus, the electronic device, and the storage medium are similar to those about the method embodiment, and technical descriptions and beneficial effects are similar to those of the corresponding method embodiment. Due to the space limitation, references can be made to the records in the method embodiments, and elaborations are omitted herein. Technical details undisclosed in the embodiments of the image conversion apparatus, the electronic device, and the storage medium in the application are understood with reference to the descriptions about the method embodiment of the application.

[00233] 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 is included in at least one embodiment of the application. Therefore, "in one embodiment" or "in an embodiment" mentioned throughout the specification does not always refer to the same embodiment. In addition, these specific features, structures or characteristics may be combined in one or more embodiments freely as appropriate. It is to be understood that, in each embodiment of the application, the magnitude of the sequence number of each process does not mean an execution sequence and the execution sequence of each process should be determined by its function and an internal logic and should not form any limit to an implementation process of the embodiments of the application. The sequence numbers of the embodiments of the application are adopted not to represent superiority-inferiority of the embodiments but

only for description. It is to be noted that terms "include" and "contain" or any other variant 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. Under the condition of no more limitations, an element defined by the statement "including a/an....." does not exclude existence of the same other elements in a process, method, object, or device including the element.

[00234] In some embodiments provided by the application, 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, and for example, division of the units is only logic function division, and other division manners may be adopted 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 and mechanical or adopt other forms.

[00235] The units described as separate parts may or may not be physically separated, and parts displayed as units may or may not be physical units, and namely may be located in the same place, or may also be distributed to multiple network units. Part of all of the units may be selected according to a practical requirement to achieve the purposes of the solutions of the embodiments.

[00236] In addition, each function unit in each embodiment of the application may be integrated into a processing unit, each unit may also serve as an independent unit and two or more than two units may also be integrated into a unit. The integrated unit may be implemented in a hardware form and may also be implemented in form of hardware and software function unit. Those of ordinary skill in the art should know 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,

and the program is executed to execute the steps of the method embodiment. 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 disc.

[00237] Or, the integrated unit of the application may also be stored in a computer-readable storage medium when 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 embodiments of the application substantially or parts making contributions to the conventional art may be embodied in form of software product. The computer software product is stored in a storage medium, including a plurality of instructions configured to enable an electronic 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 application. 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 disc. The above is only the specific implementation mode of the application and not intended to limit the scope of protection of the application. Any variations or replacements apparent to those skilled in the art within the technical scope disclosed by the application shall fall within the scope of protection of the application. Therefore, the scope of protection of the application shall be subject to the scope of protection of the claims.

CLAIMS

1. An image conversion method, comprising:

determining a first domain corresponding to a first image using an image conversion network;

determining a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image; and

converting a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.

2. The method of claim 1, wherein the determining a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image comprises:

determining a feature class shared by the first image and the preset image in each layer of the image conversion network based on the first domain and the second domain; and

executing the following steps taking the each layer as a target layer:

determining a first output feature of the first image in the target layer and a second output feature of the preset image in the target layer based on the feature class shared in the target layer respectively; and

determining the feature mapping relationship between the first image and the preset image in the target layer based on the first output feature and the second output feature.

3. The method of claim 2, wherein the determining a feature class shared by the first image and the preset image in each layer of the image conversion network based on the first domain and the second domain comprises:

determining an abstract content feature class shared by the first image and the preset image based on the first domain and the second domain;

determining a concrete content feature class shared by the first image and the preset image in each layer; and

obtaining the feature class in the each layer based on the abstract content feature class and the concrete content feature class shared in the each layer.

4. The method of claim 2 or 3, wherein the determining a first output feature of the first image in the target layer based on a feature class shared in the target layer comprises:

determining an encoder feature of the feature class shared in the target layer in the first image using a content encoder of the image conversion network;

inputting the encoder feature in the target layer to a generator of the image conversion network to determine a generator feature in the target layer; and

fusing the generator feature in the target layer and the encoder feature in the target layer to obtain the first output feature of the first image in the target layer.

5. The method of claim 4, wherein the determining an encoder feature of the feature class shared in the target layer in the first image using a content encoder of the image conversion network comprises:

determining an abstract content feature belonging to an abstract content feature class in the feature class shared in the target layer and a concrete content feature belonging to a concrete content feature class in the feature class shared in the target layer in the first image using the content encoder;

determining an initial hidden state based on the abstract content feature; and

updating the concrete content feature in the target layer based on the initial hidden state to obtain the encoder feature in the target layer.

6. The method of claim 4 or 5, wherein the inputting the encoder feature in the target layer to a generator of the image conversion network to determine a generator feature in the target layer comprises:

determining mask information of the first image in the target layer based on a hidden state in the target layer and the encoder feature in the target layer; and

inputting the concrete content feature in the target layer and the mask information in the target layer to the generator to generate the generator feature in the target layer; and

wherein the fusing the generator feature in the target layer and the encoder feature in the target layer to obtain the first output feature of the first image in the target layer comprises:

fusing the encoder feature in the target layer and the generator feature in the target layer based on the mask information in the target layer to obtain the first output feature of the first image in the target layer.

7. The method of claim 6, wherein the determining mask information of the first image in the target layer based on a hidden state in the target layer and the encoder feature in the target layer comprises:

in a case where the target layer is a first layer, determining the abstract content feature as a hidden state in the first layer; or in a case where the target layer is an i -th layer, determining a hidden state in the i -th layer based on a hidden state in an $(i-1)$ th layer and a concrete content feature in the i -th layer, wherein i is an integer greater than 1; and

determining mask information of the first image in a $(j+1)$ th layer based on a hidden state in a j -th layer and a concrete content feature in the $(j+1)$ th layer, wherein j is an integer

greater than or equal to 1.

8. The method of claim 6 or 7, wherein the inputting the concrete content feature in the target layer and the mask information in the target layer to the generator to generate the generator feature in the target layer comprises:

performing regularization processing on the mask information in the target layer to obtain sparse mask information in the target layer;

acquiring a target content feature matched with the sparse mask information in the target layer from the concrete content feature in the target layer; and

inputting the target content feature in the target layer to the generator to generate the generator feature in the target layer.

9. The method of any one of claims 6-8, wherein the fusing the encoder feature in the target layer and the generator feature in the target layer based on the mask information in the target layer to obtain the first output feature of the first image in the target layer comprises:

adjusting the encoder feature in the target layer based on the mask information in the target layer to obtain a first adjusted feature in the target layer;

adjusting the generator feature in the target layer based on adversarial information of the mask information in the target layer to obtain a second adjusted feature in the target layer; and

fusing the first adjusted feature in the target layer and the second adjusted feature in the target layer to obtain the first output feature in the target layer.

10. The method of any one of claims 1-9, wherein the converting a picture of the first image from the first domain to the second domain based on the feature mapping relationship to

obtain a second image further comprises:

determining a style feature of the preset image in each layer in the image conversion network using a style encoder in the image conversion network;

determining the style feature and corresponding feature mapping relationship in any layer in the image conversion network as a feature group; and

inputting a feature group in the each layer in the image conversion network to the generator of the image conversion network to obtain the second image.

11. The method of any one of claims 1-10, wherein the image conversion network is obtained by training a network to be trained based on a first sample image of which a picture belongs to a first sample domain and a second sample image of which a picture belongs to a second sample domain, and a training process of the network to be trained comprises:

acquiring the first sample image and second sample image which are correlated with a picture;

determining, using a trained content encoder in the network to be trained, a first sample feature of the first sample image in each layer of the trained content encoder and a second sample feature of the second sample image in the each layer;

determining a sample feature mapping relationship between the first sample image and the second sample image in each layer of the trained content encoder based on the first sample feature in the each layer and the second sample feature in the each layer;

inputting the sample feature mapping relationship in the each layer and a sample style feature of the second sample image to a first generator of the network to be trained to obtain a converted sample image after the picture is converted from the first sample domain to the second sample domain;

inputting the sample style feature of the second sample image and the second sample feature in the each layer to a second generator of the network to be trained to obtain a first

reconstructed sample image after the picture of the second sample image is reconstructed; and

adjusting a network parameter of the network to be trained based on a loss of the converted sample image and a loss of the first reconstructed sample image to obtain the image conversion network.

12. The method of claim 11, wherein the acquiring the first sample image and second sample image which are correlated with a picture comprises:

acquiring a preset noise and a domain label set at least including the first sample domain and the second sample domain; and

inputting the preset noise and the domain label set to a trained adversarial network to generate the first sample image and second sample image which have matched semantic information.

13. The method of claim 11 or 12, before the determining, using a trained content encoder in the network to be trained, a first sample feature of the first sample image in each layer of the trained content encoder and a second sample feature of the second sample image in the each layer, further comprising:

training a content encoder to be trained based on the first sample image and the second sample image to obtain the trained content encoder.

14. The method of claim 13, wherein the training a content encoder to be trained based on the first sample image and the second sample image to obtain the trained content encoder comprises:

determining a first predicted content feature of the first sample image and a second predicted content feature of the second sample image using the content encoder to be trained;

inputting the first predicted content feature, a sample style feature of the first sample image, and the first sample domain to a decoder to reconstruct the first sample image to obtain a second reconstructed sample image;

determining a first loss between the second reconstructed sample image and the first sample image;

determining a second loss between the second predicted content feature and the first predicted content feature; and

adjusting a network parameter of the content encoder to be trained based on the first loss and the second loss to obtain the trained content encoder.

15. The method of any one of claims 11-14, wherein the adjusting a network parameter of the network to be trained based on a loss of the converted sample image and a loss of the first reconstructed sample image to obtain the image conversion network comprises:

determining the loss of the converted sample image based on a content feature of the converted sample image and a content feature of the first sample image;

determining the loss of the first reconstructed sample image based on a style feature of the first reconstructed sample image and the style feature of the second sample image;

determining an adversarial loss between the first reconstructed sample image and the second sample image; and

adjusting the network parameter of the network to be trained based on the loss of the converted sample image, the loss of the first reconstructed sample image, and the adversarial loss to obtain the image conversion network.

16. An image conversion apparatus, comprising:

a first determination module, configured to determine a first domain corresponding to

a first image using an image conversion network;

a second determination module, configured to determine a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image; and

a first conversion module, configured to convert a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.

17. The apparatus of claim 16, wherein the second determination module comprises:

a first determination submodule, configured to determine a feature class shared by the first image and the preset image in each layer of the image conversion network based on the first domain and the second domain;

a second determination submodule, configured to, taking each layer as a target layer, determine a first output feature of the first image in the target layer and a second output feature of the preset image in the target layer based on the feature class shared in the target layer respectively; and

a third determination submodule, configured to determine the feature mapping relationship between the first image and the preset image in the target layer based on the first output feature and the second output feature.

18. A computer storage medium having a computer-executable instruction stored therein, wherein when executed, the computer-executable instruction is capable of implementing an image conversion method, wherein the method comprises:

determining a first domain corresponding to a first image using an image conversion network;

determining a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image; and

converting a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.

19. An electronic device, comprising a memory and a processor, wherein a computer-executable instruction is stored in the memory, wherein when executing the computer-executable instruction in the memory, the processor is capable of implementing an image conversion method, wherein the method comprises:

determining a first domain corresponding to a first image using an image conversion network;

determining a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image; and

converting a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.

20. A computer program, comprising a computer-readable code, wherein when executed in an electronic device, a processor in the electronic device is capable of implementing an image conversion method, wherein the method comprises:

determining a first domain corresponding to a first image using an image conversion network;

determining a feature mapping relationship between the first image and a preset image in each layer of the image conversion network based on the first domain and a second domain corresponding to the preset image; and

converting a picture of the first image from the first domain to the second domain based on the feature mapping relationship to obtain a second image.