

# **TEXTURE GENERATION METHOD, APPARATUS AND DEVICE, AND COMPUTER READABLE STORAGE MEDIUM**

## **TECHNICAL FIELD**

[ 0001]       The disclosure relates to the technical field of Artificial Intelligence (AI), and in particular, to a texture generation method, apparatus and device, and a computer readable storage medium.

## **BACKGROUND**

[ 0002]       AI refers to a theory, method, technology and application system that uses a digital computer or a machine controlled by the digital computer to simulate, extend and expand human intelligence, perceive environment, acquire knowledge, and use knowledge to obtain a best result. Three-dimension (3D) object reconstruction is an important branch of AI. In practical engineering applications, it is necessary to generate texture data to reconstruct a specific 3D model. It is difficult to obtain an accurate texture feature on the basis of a single image in traditional technology.

## **SUMMARY**

[ 0003]       Embodiments of the disclosure provide a texture generation method, apparatus and device, and a computer readable storage medium, which can improve the authenticity of texture parameters in a texture feature map.

[ 0004]       Technical solutions of the embodiments of the disclosure are implemented in the following manner.

[ 0005]       The embodiments of the disclosure provide a texture generation method, including the following operations.

- [ 0006]        A query feature map, a key feature map, and a value feature map of an image to be transformed are acquired. The query feature map includes a plurality of pixel points to be queried in a texture space. The key feature map is used for representing a semantic feature of the image to be transformed. The value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried. The attention feature map is determined by the query feature map and the key feature map.
- [ 0007]        A target texture feature map is generated from the query feature map, the key feature map, and the value feature map on the basis of an attention mechanism. The target texture feature map includes the texture parameter corresponding to each of the pixel points to be queried.
- [ 0008]        In some embodiments, the operation that a target texture feature map is generated from the query feature map, the key feature map, and the value feature map on the basis of an attention mechanism includes the following operations.
- [ 0009]        Feature extraction processing at different scales is performed on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales. The input feature map includes the query feature map, the key feature map, and the value feature map.
- [ 0010]        An input feature corresponding to each of the scales are input into a corresponding transforming module to obtain an output feature corresponding to each of the scales.
- [ 0011]        The output features corresponding to all of the scales are fused to obtain the target texture feature map.
- [ 0012]        In some embodiments, the operation that the feature extraction processing at different scales is performed on an input feature map of the attention mechanism to obtain an input feature corresponding to a plurality of scales includes the following operations.
- [ 0013]        Downsampling processing is performed on the input feature map for a plurality

of times to obtain an input feature corresponding to a preset scale.

[ 0014]       Upsampling processing is performed on the input feature corresponding to the preset scales for a plurality of times to obtain input features corresponding to the plurality of scales. The number of times of the downsampling processing is the same as that of the upsampling processing.

[ 0015]       In some embodiments, the operation that an input feature corresponding to each of the scales is input into a corresponding transforming module to obtain an output feature corresponding to each of the scales includes the following operations.

[ 0016]       For each scale, the query feature map and a position code of the query feature map are fused to obtain a first input feature.

[ 0017]       The key feature map and a position code of the key feature map are fused to obtain a second input feature.

[ 0018]       An intermediate feature map is obtained on the basis of the first input feature, the second input feature, and the value feature map.

[ 0019]       The output feature corresponding to the scale is determined on the basis of the intermediate feature map and a residual network.

[ 0020]       In some embodiments, the operation that the output features corresponding to all of the scales are fused to obtain the target texture feature map includes the following operations.

[ 0021]       The output features corresponding to all of the scales are fused to obtain a multi-scale output feature.

[ 0022]       The target texture feature map is determined on the basis of the multi-scale output feature.

[ 0023]       In some embodiments, the operation that the output features corresponding to all of the scales are fused to obtain a multi-scale output feature includes the following operations.

[ 0024] Downsampling fusion processing is performed on the output feature corresponding to each of the scales in turn according to an order from the largest scale to the smallest scale of the plurality of scales to obtain a downsampling fusion feature corresponding to each of the scales. The downsampling fusion processing is used for fusing the output feature corresponding to the scale and a downsampling fusion feature obtained from a last downsampling fusion processing to obtain a downsampling intermediate feature, and performing downsampling processing on the downsampling intermediate feature to obtain a downsampling fusion feature corresponding to the scale.

[ 0025] Upsampling fusion processing is performed on the output feature corresponding to each of the scales in turn according to an order from the smallest scale to the largest scale of the plurality of scales to obtain an upsampling fusion feature corresponding to each of the scales. The upsampling fusion processing is used for fusing the output feature corresponding to the scale and an upsampling fusion feature obtained from a last downsampling fusion processing to obtain an upsampling intermediate feature, and performing upsampling processing on the upsampling intermediate feature to obtain an upsampling fusion feature corresponding to the scale.

[ 0026] The upsampling fusion processing of the smallest scale is used for performing upsampling processing on a downsampling fusion feature corresponding to the smallest scale, and fusing the output feature corresponding to the smallest scale, to obtain an upsampling fusion feature corresponding to the smallest scale. An upsampling fusion feature corresponding to the largest scale is a multi-scale output feature.

[ 0027] In some embodiments, in the case where the multi-scale output feature includes a pixel feature map, the value feature map includes an original pixel feature of the image to be transformed. The operation that the target texture feature map is determined on the basis of the multi-scale output feature includes the following operation.

[ 0028] The pixel feature map is taken as the target texture feature map.

[ 0029] In some embodiments, in the case where the multi-scale output feature includes a texture flow feature map, the value feature map includes a pixel distribution feature of the image to be transformed. The operation that the target texture feature map

is determined on the basis of the multi-scale output feature includes the following operation.

[ 0030]        The target texture feature map is generated on the basis of the texture flow feature map and the image to be transformed.

[ 0031]        In some embodiments, in the case where the multi-scale output feature includes the pixel feature map, the texture flow feature map, and a confidence feature map, the value feature map includes an original pixel feature and a pixel distribution feature of the image to be transformed. The operation that the target texture feature map is determined on the basis of the multi-scale output feature includes the following operations.

[ 0032]        Confidence information of each of the pixel points to be queried in the texture space is determined on the basis of the confidence feature map. The confidence information includes a first confidence of the pixel point to be queried corresponding to the pixel feature map and a second confidence of the pixel point to be queried corresponding to the texture flow feature map.

[ 0033]        The pixel feature map and the texture flow feature map are fused on the basis of the first confidence and the second confidence of each of the pixel points to be queried, to obtain the target texture feature map.

[ 0034]        In some embodiments, the operation that the pixel feature map and the texture flow feature map are fused on the basis of the first confidence and the second confidence of each of the pixel points to be queried to obtain the target texture feature map includes the following operations.

[ 0035]        A first texture parameter corresponding to each of the pixel points to be queried is determined on the basis of the pixel feature map.

[ 0036]        A second texture parameter corresponding to each of the pixel points to be queried is determined on the basis of the texture flow feature map and the image to be transformed.

- [ 0037] For each of the pixel points to be queried, weighted fusion is performed on the first texture parameter and the second texture parameter on the basis of the first confidence and the second confidence respectively to obtain a texture parameter corresponding to the pixel point to be queried.
- [ 0038] In some embodiments, the texture generation method provided by the above-mentioned embodiments is implemented by a trained texture generation model. A training process of the texture generation model includes the following operations.
- [ 0039] A sample image is acquired, and the sample image is input into a texture generation model to be trained to obtain a predicted texture image corresponding to the sample image.
- [ 0040] The predicted texture image corresponding to the sample image is rendered to obtain a predicted rendered image corresponding to the sample image.
- [ 0041] A loss value of the texture generation model to be trained is determined on the basis of the sample image and the predicted rendered image.
- [ 0042] A parameter of the texture generation model to be trained is adjusted according to the loss value to obtain the trained texture generation model.
- [ 0043] In some embodiments, the loss value of the texture generation model to be trained includes a first loss value.
- [ 0044] The operation that a loss value of the texture generation model to be trained is determined on the basis of the sample image and the predicted rendered image includes the following operations. A first character feature in the sample image and a second character feature in the predicted rendered image are respectively extracted on the basis of a person ReID model. The first loss value is determined on the basis of a feature distance between the first character feature and the second character feature.
- [ 0045] The operation that a parameter of the texture generation model to be trained is adjusted according to the first loss value to obtain the trained texture generation model includes the following operation. The parameter of the texture generation model to be

trained is adjusted on the basis of at least the first loss value to obtain the trained texture generation model.

[ 0046] In some embodiments, the loss value of the trained texture generation model to be trained includes a second loss value.

[ 0047] The operation that a loss value of the texture generation model to be trained is determined on the basis of the sample image and the predicted rendered image includes the following operations. A first character feature in the sample image and a second character feature in the predicted rendered image are respectively extracted on the basis of a person ReID model. A first segmentation result corresponding to the sample image is acquired, the first segmentation result being used for representing first distribution information of each body part in the sample image. A second segmentation result corresponding to the predicted rendered image is acquired, the second segmentation result being used for representing second distribution information of each body part in the predicted rendered image. For each body part, a part loss value corresponding to the body part is determined on the basis of a feature distance between a first part feature and a second part feature of the body part, the first part feature being determined on the basis of the first character feature and first part information of the body part, the second part feature being determined on the basis of the second character feature and second part information of the body part. The second loss value is determined on the basis of the part loss value corresponding to the each body part.

[ 0048] The operation that a parameter of the texture generation model to be trained is adjusted to obtain the trained texture generation model according to the loss value includes the following operation. The parameter of the texture generation model to be trained is adjusted on the basis of at least the second loss value to obtain the trained texture generation model.

[ 0049] In some embodiments, the loss value of the texture generation model to be trained includes a third loss value. The method further includes the following operations. A part texture image corresponding to a target body part in the predicted texture image is acquired. A standard texture image corresponding to the target body part is acquired, the standard texture image being a fusion image of part texture images of different characters

regarding the target body part. The third loss value is determined on the basis of a structural similarity index between the part texture image and the standard texture image.

[ 0050]        The operation that a parameter of the texture generation model to be trained is adjusted according to the loss value to obtain the trained texture generation model includes the following operation. The parameter of the texture generation model to be trained is adjusted on the basis of at least the third loss value to obtain the trained texture generation model.

[ 0051]        The embodiments of the disclosure provide a texture generation apparatus. The apparatus includes an acquisition module and a generation module.

[ 0052]        The acquisition module is configured to acquire a query feature map, a key feature map, and a value feature map of an image to be transformed. The query feature map includes a plurality of pixel points to be queried in a texture space. The key feature map is used for representing a semantic feature of the image to be transformed. The value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried. The attention feature map is determined by the query feature map and the key feature map.

[ 0053]        The generation module is configured to generate a target texture feature map from the query feature map, the key feature map, and the value feature map on the basis of an attention mechanism. The target texture feature map includes the texture parameter corresponding to each of the pixel points to be queried.

[ 0054]        The embodiments of the disclosure provide a texture generation device, including a memory, and a processor.

[ 0055]        The memory is configured to store an executable instruction.

[ 0056]        The processor is configured to implement the texture generation method provided by the embodiments of the disclosure when executing the executable instruction stored in the memory.

[ 0057]       The embodiments of the disclosure provide a computer-readable storage medium, which stores an executable instruction for implementing the texture generation method provided by the embodiments of the disclosure when being executed by the processor.

[ 0058]       The embodiments of the disclosure have the following beneficial effects.

[ 0059]       In the embodiments of the disclosure, in a process of acquiring the target texture feature map on the basis of the query feature map, the key feature map, and the value feature map, the texture parameter of each of the pixel points to be queried in the target texture feature map can consider global information of the image to be transformed because of the adoption of the attention mechanism, so as to improve the authenticity of the texture parameter in the texture feature map, and provide a texture base for subsequent rendering of a corresponding 3D grid.

#### BRIEF DESCRIPTION OF THE DRAWINGS

[ 0060]       FIG. 1A is an optional flowchart of a texture generation method provided by embodiments of the disclosure.

[ 0061]       FIG. 1B is an optional feature relationship diagram of an attention mechanism provided by the embodiments of the disclosure.

[ 0062]       FIG. 2A is an optional flowchart of a texture generation method provided by embodiments of the disclosure.

[ 0063]       FIG. 2B is a structural diagram of an optional attention module provided by the embodiments of the disclosure.

[ 0064]       FIG. 3 is an optional flowchart of a texture generation method provided by the embodiments of the disclosure.

[ 0065]       FIG. 4 is an optional flowchart of a texture generation method provided by the embodiments of the disclosure.

- [ 0066] FIG. 5 is an optional flowchart of a training method for a texture generation model provided by the embodiments of the application.
- [ 0067] FIG. 6 is an optional flowchart of a first loss value acquisition method provided by the embodiments of the application.
- [ 0068] FIG. 7 is an optional flowchart of a second loss value acquisition method provided by the embodiments of the application.
- [ 0069] FIG. 8 is an optional flowchart of a third loss value acquisition method provided by the embodiments of the application.
- [ 0070] FIG. 9 is an another optional flowchart of a training method for a texture generation model provided by the embodiments of the application.
- [ 0071] FIG. 10 is a schematic diagram of a texture generation method based on a Convolutional Neural Network (CNN) provided by the embodiments of the disclosure.
- [ 0072] FIG. 11 is a schematic diagram of a model architecture of a texture generation model provided by the embodiments of the disclosure.
- [ 0073] FIG. 12 is a schematic diagram of rendering results of different texture feature maps provided by the embodiments of the disclosure.
- [ 0074] FIG. 13 is a schematic diagram of rendering results of different loss functions provided by the embodiments of the disclosure.
- [ 0075] FIG. 14 is a schematic diagram of a composition structure of a texture generation apparatus provided by the embodiments of the disclosure.
- [ 0076] FIG. 15 is a schematic diagram of a hardware entity of a texture generation device provided by the embodiments of the disclosure.

## DETAILED DESCRIPTION

[ 0077]       The following specifically describes technical solutions of the present disclosure in detail with reference to accompanying drawings. The following specific embodiments may be combined. The same or similar concepts or processes will not be elaborated in some embodiments.

[ 0078]       It is to be noted that, in the embodiments of the disclosure, "first", "second" and the like are adopted to distinguish between similar objects and not intended to describe a target sequence or order. In addition, the technical solutions recorded in the embodiments of the disclosure may be freely combined without conflicts.

[ 0079]       FIG. 1A is an optional flowchart of a texture generation method provided by the embodiments of the disclosure, which will be described in combination with the steps shown in FIG. 1A.

[ 0080]       At S101, a query feature map, a key feature map, and a value feature map of an image to be transformed are acquired. The query feature map includes a plurality of pixel points to be queried in a texture space. The key feature map is used for representing a semantic feature of the image to be transformed. The value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried. The attention feature map is determined by the query feature map and the key feature map.

[ 0081]       In some embodiments, the query feature map includes a plurality of pixel points to be queried in a texture space. Each of the pixel points to be queried corresponds to a vertex in a 3D body grid to be rendered. FIG. 1B is an optional feature relationship diagram of an attention mechanism provided by the embodiments of the disclosure. For the pixel point to be queried of a query feature map 11 in FIG. 1B, taking the pixel point O1 to be queried marked with an asterisk in the figure as an example, the pixel point O1 to be queried corresponds to one vertex O2 in a 3D body grid 16.

[ 0082]       In some embodiments, semantic segmentation may be performed on the image to be transformed through a two-dimensional semantic segmentation model to obtain a key feature map which is used for representing a semantic feature of the image to be transformed. As shown in the key feature map 12 in FIG. 1B, the key feature map 12 has

included a semantic feature of each region (pixel point) in the image to be transformed. Exemplarily, the key feature map 12 in FIG. 1B has segmented the image to be transformed into semantic parts, such as a head, a trunk, upper arms, arms, thighs and calves.

[ 0083] It is to be noted that the attention feature map corresponding to each of the pixel points to be queried may be obtained on the basis of the query feature map and the key feature map. As shown in the attention feature map 13 in FIG. 1B, taking the above-mentioned pixel point O1 to be queried as an example, an attention distribution situation as shown in the attention feature map 13 may be obtained after a correlation operation is performed between the query feature map and the key feature map. The pixel point O1 to be queried is located at an arm part in the 3D body grid, so the attention feature in the obtained attention feature map 13 is also mainly distributed in a region with a semantic feature of "arm" in the key feature map.

[ 0084] In some embodiments, the value feature map is used for determining, in combination with the attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried. Referring to FIG. 1B, taking a value feature map 14 being an image to be transformed as an example, a target pixel value corresponding to the pixel point O1 to be queried may be obtained on the basis of a weight corresponding to each of the pixel points in the attention feature map 13 and a pixel value of each pixel point in the value feature map 14. The target pixel value is taken as a texture parameter of a pixel point O3 to be queried in a target texture map 15. The pixel point O3 to be queried in the target texture map 15 corresponds to the pixel point O1 to be queried in the query feature map 11.

[ 0085] At S102, a target texture feature map is generated from the query feature map, the key feature map, and the value feature map on the basis of an attention mechanism. The target texture feature map includes a texture parameter corresponding to each of the pixel points to be queried.

[ 0086] In some embodiments, the query feature map is taken as query ( $Q \in \mathbb{R}^{v \times d}$ ) of the attention mechanism. The key feature map is taken as key ( $K \in \mathbb{R}^{h \times d}$ ) of the

attention mechanism. The value feature map is taken as value ( $V \in \mathbb{R}^{hw \times c}$ ) of the attention mechanism. The target texture feature map ( $O \in \mathbb{R}^{vu \times d}$ ) is obtained through the attention mechanism. The (v, u) is the height and the width of an output texture space (UV space); the (h, w) are the height and width of an image to be transformed; d, c are respectively feature dimensions of K and V.

[ 0087] In some embodiments, the feature parameter may be a pixel value.

[ 0088] In the embodiments of the disclosure, in a process of acquiring the target texture feature map on the basis of the query feature map, the key feature map, and the value feature map, the texture parameter of each of the pixel points to be queried in the target texture feature map can consider global information of the image to be transformed because of the adoption of the attention mechanism, so as to improve the authenticity of the texture parameter in the texture feature map on the basis of a single image, and provide a texture base for subsequent rendering of a corresponding 3D grid.

[ 0089] FIG. 2A is an optional flowchart of a texture generation method provided by the embodiments of the disclosure. S102 in FIG. 1A may include S201 to S203, which will be described in combination with the steps shown in FIG. 2A.

[ 0090] At S201, feature extraction processing at different scales is performed on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales. The input feature map includes the query feature map, the key feature map, and the value feature map.

[ 0091] In some embodiments, in order to obtain deep features of the query feature map, the key feature map, and the value feature map, and query feature maps, key feature maps, and value feature maps corresponding to at least two scales, for ease of expression, embodiments hereinafter will replace the query feature map, the key feature map, and the value feature map with an input feature map because the query feature map, the key feature map, and the value feature map are all input features of the attention mechanism. It is to be noted that the query feature map is independent of the image to be transformed, i.e., the same query feature map may be adopted for different images to be transformed, so the query feature map corresponding to each scale may be set in advance, so as to

avoid repeated feature extraction of the query feature map.

[ 0092]        In some embodiments, downsampling processing may be performed on the input feature map for a plurality of times to obtain input features corresponding to a plurality of scales. The feature obtained from each downsampling processing is an input feature corresponding to one scale.

[ 0093]        In some embodiments, for the input features corresponding to the plurality of scales, as long as the above-mentioned input features corresponding to the plurality of scales are obtained, the embodiments of the disclosure may adopt any number of times of downsampling processing, any number of times of upsampling processing, and processing sequence.

[ 0094]        In some embodiments, the feature extraction processing at different scales may be implemented on the input feature map of the attention mechanism through S2011, so as to obtain input features corresponding to a plurality of scales.

[ 0095]        At S2011, downsampling processing is performed on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale; upsampling processing is performed on the input feature corresponding to the preset scale for a plurality of times to obtain input features corresponding to the plurality of scales, the number of times of the downsampling processing being the same as that of the upsampling processing.

[ 0096]        In some embodiments, the feature obtained by each upsampling processing is an input feature corresponding to one scale.

[ 0097]        For example, in the case where the plurality of times of downsampling processing is three times of downsampling processing and the plurality of times of upsampling processing is three times of upsampling processing, for an input feature of an original scale (D0), the feature of scale D1 is obtained after first downsampling processing is performed. For the feature of scale D1, the feature of scale D2 is obtained after second downsampling processing is performed. For the feature of scale D2, the feature of scale D3, i.e., the input feature corresponding to the above-mentioned preset scale, is obtained after third downsampling processing is performed. For the feature of

scale D3, the feature of scale U2 is obtained after first upsampling processing is performed. For the feature of scale U2, the feature of scale U1 is obtained after second upsampling processing is performed. For the feature of scale U1, the feature of scale U0 is obtained after third upsampling processing is performed. The feature of scale U2, the feature of scale U1, and the feature of scale U0 are the input features corresponding to three scales.

[ 0098]        It is to be noted that scale D0 and scale U0, scale D1 and scale U1, and scale D2 and scale U2 may be completely or partially the same, or may be completely different. When the scale D0 and scale U0, scale D1 and scale U1, and scale D2 and scale U2 are completely the same, and when both the number of times of the downsampling processing and the number of times of the upsampling processing are N, the scale variation of the n-th downsampling processing is the same as the scale variation of the (N-n+1)-th upsampling process.

[ 0099]        At S202, an input feature corresponding to each of the scales is input into a corresponding transforming module to obtain output features corresponding to each of the scales.

[ 00100]       In some embodiments, for each scale, the query feature map as Q, the key feature map as K, and the value feature map as V of the current scale are input into a transforming module corresponding to the current scale, and 0 output by the transforming module is taken as an output feature corresponding to the scale.

[ 00101]       In some embodiments, the above-mentioned operation that the input feature corresponding to each of the scales is input into the corresponding transforming module to obtain the output feature corresponding to each of the scales may be implemented through S2021.

[ 00102]       At S2021, for each of the scales, the query feature map and a position code of the query feature map are fused to obtain a first input feature; the key feature map and a position code of the key feature map are fused to obtain a second input feature; an intermediate feature map is obtained on the basis of the first input feature, the second input feature, and the value feature map; the output feature corresponding to the scale is

determined on the basis of the intermediate feature map and a residual network.

[ 00103]        The above-mentioned output feature may be represented as Formula (1-1):

$$O: f_{\text{res-MLP}}(f_{\text{Attn}}(Q + E_Q, K + E_K, V)) \quad \text{Formula (1-1)}$$

where O is the output feature, Q is the query feature map,  $E_Q$  is the position code of the query feature map,  $(Q + E_Q)$  is a first input feature, K is the key feature map,  $E_K$  is the position code of the key feature map,  $(K + E_K)$  is the second input feature, V is a value feature map,  $f_{\text{Attn}}$  is an attention module, and  $f_{\text{res-MLP}}$  is a residual network.

[ 00104]        FIG. 2B shows a structural diagram of an attention module. The attention module may obtain, on the basis of the value feature map V, the intermediate feature map by fusing the first input feature  $(Q + E_Q)$  of the position code of the query feature map and the second input feature  $(K + E_K)$  of the position code of the value feature map, and then obtain the output feature O corresponding to the current scale through the residual network by fusing the intermediate feature and the intermediate feature which is processed by a Multilayer Perceptron (MLP).

[ 00105]        At S203, the output features corresponding to all of the scales are fused to obtain the target texture feature map.

[ 00106]        In some embodiments, in the case where the value feature map includes an original pixel feature of an image to be transformed, the output feature corresponding to each of the scales obtained on the basis of the attention mechanism also includes a pixel feature. Therefore, the target texture feature map may be obtained after the output features corresponding to all of the scales are fused.

[ 00107]        In some embodiments, in the case where the value feature map includes a pixel distribution feature of an image to be transformed, the output feature corresponding to each of the scales obtained on the basis of the attention mechanism also includes the pixel distribution feature. Therefore, after the output features corresponding to all of the scales are fused, not only the output features corresponding to all of the scales need to be fused to obtain a multi-scale output feature, but also a target texture feature map needs to

be determined in combination with the multi-scale output feature and the image to be transformed. That is, the above-mentioned operation that the output features corresponding to all of the scales are fused to obtain the target texture feature map may be implemented through S2031 and S2032.

[ 00108] At S2031, the output features corresponding to all of the scales are fused to obtain a multi-scale output feature.

[ 00109] At S2032, the target texture feature map is determined on the basis of the multi-scale output feature.

[ 00110] In some embodiments, in the case where the multi-scale output feature includes a pixel feature map, the value feature map includes an original pixel feature of the image to be transformed. The operation that the target texture feature map is determined on the basis of the multi-scale output feature includes that the pixel feature map is taken as the target texture feature map.

[ 00111] In some embodiments, in the case where the multi-scale output feature includes a texture flow feature map, the value feature map includes a pixel distribution feature of the input image. The operation that the target texture feature map is determined on the basis of the multi-scale output feature includes that the target texture feature map is generated on the basis of the texture flow feature map and the image to be transformed.

[ 00112] In some embodiments, the multi-scale output feature may include both the pixel feature map and the texture flow feature map. For example, the multi-scale output feature is  $v*u*C$ , where  $(v, u)$  is the height and width of an output texture space (UV space),  $C$  is a feature corresponding to each of the pixel point to be queried in the texture space. In the embodiments of the disclosure,  $C$  may be segmented into  $C1$  and  $C2$  to obtain a pixel feature map of  $v*u*C1$  and a texture flow feature map of  $v*u*C2$ . Later, a corresponding first texture feature map to be fused is obtained through the pixel feature map, and a corresponding second texture feature map to be fused is obtained through the texture flow feature map. The first texture feature map to be fused and the second texture feature map to be fused are fused to obtain the target texture feature map on the basis of a preset fusion weight.

[ 00113] In the embodiments of the disclosure, the feature distribution situation of each feature map at different scales may be acquired because the output feature corresponding to each scale is obtained by acquiring the input features of different scales (including the query feature map, the key feature map, and the value feature map) and combining corresponding transforming modules, so the target texture feature map obtained on the basis of the output features corresponding to the above-mentioned different scales may reflect real texture information more accurately.

[ 00114] FIG. 3 is an optional flowchart of a texture generation method provided by the embodiments of the disclosure. S2031 in FIG. 2A may include S301 to S302, which will be described in combination with the steps shown in FIG. 3.

[ 00115] At S301, downsampling processing is performed on the output feature corresponding to each of the scales in turn according to an order from the largest scale to the smallest scale of the plurality of scales to obtain a downsampling fusion feature corresponding to each of the scale .

[ 00116] In some embodiments, the downsampling fusion processing is used for fusing the output feature corresponding to the scale and the downsampling fusion feature obtained from a last downsampling fusion processing to obtain a downsampling intermediate feature, and performing downsampling processing on the downsampling intermediate feature to obtain a downsampling fusion feature corresponding to the scale.

[ 00117] At S302, upsampling fusion processing is performed on the output feature corresponding to each of the scales in turn according to an order from the smallest scale to the largest scale of the plurality of scales to obtain corresponding upsampling fusion features corresponding to each of the scales, the upsampling fusion feature corresponding to the largest scale being the multi-scale output feature.

[ 00118] In some embodiments, the upsampling fusion processing is used for performing upsampling processing on the upsampling fusion feature obtained from a last upsampling fusion process to obtain an upsampling intermediate feature, and fusing the upsampling intermediate feature and the output feature corresponding to the scale to obtain an upsampling fusion feature corresponding to the scale.

- [ 00119] In some embodiments, the upsampling fusion processing of the smallest scale is used for performing upsampling processing on the downsampling fusion feature corresponding to the smallest scale, and fusing the output feature corresponding to the smallest scale to obtain an upsampling fusion feature corresponding to the smallest scale. The upsampling fusion feature corresponding to the largest scale is a multi-scale output feature.
- [ 00120] Taking the plurality of scales including a first scale, a second scale, and a third scale as an example, the first scale > the second scale > the third scale according to an order from the largest scale to the smallest scale. An executing process of the above-mentioned S301 includes the operations. First, downsampling fusion processing is performed on an output feature corresponding to the first scale; then, downsampling fusion processing is performed on an output feature corresponding to the second scale; and finally, downsampling fusion processing is performed on an output feature corresponding to the third scale.
- [ 00121] During the downsampling fusion processing on the output feature corresponding to the first scale, since the first scale is the largest scale, it is not necessary to fuse the downsampling fusion feature obtained from a last downsampling fusion processing, but only to perform the downsampling processing on the output feature of the first scale to obtain the downsampling fusion feature corresponding to the first scale.
- [ 00122] During the downsampling fusion processing on the output feature corresponding to the second scale, the downsampling fusion feature corresponding to the first scale and the output feature corresponding to the second scale are fused to obtain a downsampling intermediate sample of the second scale. Then, the downsampling fusion processing is performed on the downsampling intermediate sample to obtain the downsampling fusion feature corresponding to the second scale. By analogy, during the downsampling fusion processing on the output feature corresponding to the second scale, the downsampling fusion feature corresponding to the third scale may be obtained.
- [ 00123] At this time, S301 has completed the upsampling fusion processing performed on the output features corresponding to each of the scales in turn to obtain the upsampling fusion feature corresponding to each of the scales, and transfers the

downsampling fusion feature corresponding to the smallest scale (the third scale) to S302.

[ 00124]       An executing process of the above-mentioned S302 includes the operations. First, upsampling fusion processing is performed on an output feature corresponding to the third scale; then, the upsampling fusion processing is performed on an output feature corresponding to the second scale; and finally, the upsampling fusion processing is performed on an output feature corresponding to the first scale.

[ 00125]       During the upsampling fusion processing on the output feature corresponding to the third scale, since the third scale is the smallest scale, it is necessary to perform the upsampling processing on the downsampling fusion feature corresponding to the smallest scale (the third scale) obtained in S301 to obtain an upsampling fusion feature corresponding to the third scale.

[ 00126]       During the upsampling fusion processing on the output feature corresponding to the second scale, the upsampling processing is performed on the upsampling fusion feature corresponding to the third scale to obtain an upsampling intermediate feature corresponding to the second scale, and the upsampling intermediate feature and the output feature corresponding to the second scale are fused to obtain an upsampling fusion feature corresponding to the second scale. By analogy, during the upsampling fusion processing on the output feature corresponding to the first scale, the upsampling fusion feature corresponding to the first scale may be obtained.

[ 00127]       At this time, S302 has completed the upsampling fusion processing performed on the output features corresponding to each of the scales in turn to obtain the upsampling fusion feature corresponding to each of the scales, and takes the upsampling fusion feature corresponding to the largest scale (the first scale) as the multi-scale output feature.

[ 00128]       In the embodiments of the disclosure, during fusing, for the upsampling fusion processing or the downsampling fusion processing corresponding to the current scale, not only the output feature obtained from a last fusing process is considered, but also an original output feature of the current scale is considered, so that the feature fusion effect can be improved, and the obtained multi-scale output feature may include feature

information of the output feature of each scale.

[ 00129] FIG. 4 is an optional flowchart of a texture generation method provided by the embodiments of the disclosure. S2032 in FIG. 2A may include S401 to S402, which will be described in combination with the steps shown in FIG. 4.

[ 00130] At S401, confidence information of each of the pixel points to be queried in the texture space is determined on the basis of the confidence feature map, the confidence information including a first confidence of the pixel point to be queried corresponding to the pixel feature map and a second confidence of the pixel point to be queried corresponding to the texture flow feature map.

[ 00131] In some embodiments, the above-mentioned multi-scale fusion feature not only includes the pixel feature map and the texture flow feature map, but also includes a confidence feature map carrying confidence information of each of the pixel points to be queried. For example, the multi-scale output feature is  $v*u*C$ , where  $(v, u)$  is the height and the width of an output texture space (UV space),  $C$  is a feature corresponding to each of the pixel point to be queried in the texture space. In the embodiments of the disclosure,  $C$  may be segmented into  $C_1$ ,  $C_2$ , and  $C_3$  to obtain a pixel feature map of  $v*u*C_1$ , a texture flow feature map of  $v*u*C_2$ , and a confidence feature map of  $v*u*C_3$ .

[ 00132] In some embodiments, for any pixel point to be queried, the confidence information of the pixel point to be queried may be represented as  $(K1, K2)$ .  $K1$  is a first confidence of the pixel feature map corresponding to the pixel point to be queried. The higher the first confidence, the more accurate the texture parameter representing a restored pixel point to be queried in the pixel feature map.  $K2$  is a second confidence of the texture flow feature map corresponding to the pixel point to be queried. The higher the second confidence, the more accurate the texture parameter representing a restored pixel point to be queried in the texture flow feature map.

[ 00133] At S402, the pixel feature map and the texture flow feature map are fused on the basis of the first confidence and the second confidence of each of the pixel points to be queried to obtain the target texture feature map.

[ 00134] In some embodiments, for each pixel point to be queried, the first confidence

may be compared with the second confidence. The texture parameter of the pixel point to be queried is determined on the basis of the feature map (the pixel feature map or the texture flow feature map) corresponding to a greater confidence, so as to obtain the texture parameters of all of the pixel points to be queried, i.e., the target texture feature map.

[ 00135] In some embodiments, the above-mentioned operation that the input feature corresponding to each of the scales are input into the corresponding transforming module to obtain the output feature corresponding to each of the scales may be implemented through S4021 to S4023.

[ 00136] At S4021, a first texture parameter corresponding to each of the pixel points to be queried is determined on the basis of the pixel feature map.

[ 00137] At S4022, a second texture parameter corresponding to each of the pixel points to be queried is determined on the basis of the texture flow feature map and the image to be transformed.

[ 00138] At S4023, for each of the pixel points to be queried, weighted fusion is performed on the first texture parameter and the second texture parameter on the basis of the first confidence and the second confidence respectively to obtain a texture parameter corresponding to the pixel point to be queried.

[ 00139] Taking the texture parameter being a pixel value as an example, for any pixel point to be queried, the confidence information of the pixel point to be queried is (K1, K2), then a first texture parameter P1 corresponding to the pixel point to be queried may be determined on the basis of the pixel feature map, and a first texture parameter P1 corresponding to the pixel point to be queried may be determined on the basis of the texture flow feature map and the image to be transformed. The texture parameter corresponding to the pixel point to be queried may be represented as  $P=K1*P1+K2*P2$ .

[ 00140] It is to be noted that the texture parameter may also be other texture data for rendering a 3D model, for example, at least one of brightness, material, or hardness.

[ 00141] In the embodiments of the disclosure, the first texture parameter determined by

the pixel feature map and the second texture parameter determined by the texture flow feature map are fused on the basis of the first confidence and the second confidence of the pixel point to be queried, which can improve the authenticity of the texture parameter of each pixel point to be queried.

[ 00142] FIG. 5 is an optional flowchart of a training method of a texture generation model provided by the embodiments of the application. The texture generation method in the above-mentioned embodiments is implemented by the trained texture generation model. A training process of the texture generation model may be implemented through S501 and S504, which will be described in combination with the steps shown in FIG. 5.

[ 00143] At S501, a sample image is acquired, and the sample image is input into a texture generation model to be trained to obtain a predicted texture image corresponding to the sample image.

[ 00144] At S502, the predicted texture image corresponding to the sample image is rendered to obtain a predicted rendered image corresponding to the sample image.

[ 00145] At S503, a loss value of the texture generation model to be trained is determined on the basis of the sample image and the predicted rendered image.

[ 00146] The loss value of the texture generation model to be trained may be determined by at least one of a first loss value, a second loss value, and a third loss value. Acquisition methods for different loss values are described below.

[ 00147] In some embodiments, FIG. 6 is an optional flowchart of a first loss value acquisition method provided by the embodiments of the application.

[ 00148] At S601, a first character feature in the sample image and a second character feature in the predicted rendered image are respectively extracted on the basis of a person re-identification (ReID) model.

[ 00149] At S602, the first loss value is determined on the basis of a feature distance between the first character feature and the second character feature.

[ 00150] In some embodiments, FIG. 7 is an optional flowchart of a second loss value

acquisition method provided by the embodiments of the application, which includes the following operations.

[ 00151] At S701, a first character feature in the sample image and a second character feature in the predicted rendered image are respectively extracted on the basis of a person ReID model.

[ 00152] At S702, a first segmentation result corresponding to the sample image is acquired, the first segmentation result being used for representing first distribution information of each body part in the sample image; a second segmentation result corresponding to the predicted rendered image is acquired, the second segmentation result being used for representing second distribution information of each body part in the predicted rendered image.

[ 00153] At S703, for each body part, a part loss value corresponding to the body part is determined on the basis of a feature distance between a first part feature and a second part feature of the body part, the first part feature being determined on the basis of the first character feature and first part information of the body part, the second part feature being determined on the basis of the second character feature and second part information of the body part.

[ 00154] At S704, the second loss value is determined on the basis of the part loss value corresponding to each body part.

[ 00155] In some embodiments, FIG. 8 is an optional flowchart of a third loss value acquisition method provided by the embodiments of the application, which includes the following operations.

[ 00156] At S801, a part texture image corresponding to a target body part in the predicted texture image is acquired.

[ 00157] At S802, a standard texture image corresponding to the target body part is acquired. The standard texture image is a fusion image of part texture images of different characters regarding the target body part.

- [ 00158]        At S803, the third loss value is determined on the basis of a structural similarity index between the part texture image and the standard texture image.
- [ 00159]        At S804, the parameter of the texture generation model to be trained is adjusted according to the loss value to obtain a trained texture generation model.
- [ 00160]        In some embodiments, the loss value of the texture generation model to be trained may also be determined on the above-mentioned first loss value, the second loss value, and the third loss value. The texture feature map output by the texture generation model obtained by training through a loss function may not only retain detail information of the image to be transformed, but also reduce the probability of artifacts. For body parts with complex structures, a texture feature map that accords with body features may be generated.
- [ 00161]        FIG. 9 is another optional flowchart of a training method of a texture generation model provided by the embodiments of the application. The texture generation method in the above-mentioned embodiments is implemented by the trained texture generation model. A training process of the texture generation model may also be implemented through S901 to S904, which will be described in combination with the steps shown in FIG. 9.
- [ 00162]        At S901, a sample image is acquired, and the sample image is input into a texture generation model to be trained to obtain a predicted texture image corresponding to the sample image.
- [ 00163]        At S902, the predicted texture image corresponding to the sample image is rendered to obtain a predicted rendered image corresponding to the sample image.
- [ 00164]        At S903, the first loss value, the second loss value, and the third loss value are determined on the basis of the sample image, the predicted rendered image, and the standard texture image corresponding to the target body part.
- [ 00165]        At S904, a total loss value of the texture generation model to be trained is determined on the basis of weight parameters respectively corresponding to the first loss value, the second loss value, and the third loss value.

[ 00166] At S905, the parameter of the texture generation model to be trained is adjusted according to the total loss value to obtain the trained texture generation model.

[ 00167] In the embodiments of the disclosure, the loss value of the texture generation model to be trained is determined on all of the first loss value, the second loss value, and the third loss value described above during training a texture generation image to be trained by using the sample image, so the texture feature map output by the texture generation model obtained by training through a loss function may not only retain detail information of the image to be transformed, but also reduced the probability of artifacts. For body parts with complex structures, a texture feature map that accords with body features may be generated.

[ 00168] An exemplary application of the embodiments of the disclosure in an actual application scenario will be described below. For ease of understanding, the following will describe a body as the 3D model (grid) to be rendered. The application may also be applied to other objects, such as animals, buildings, and vehicles.

[ 00169] The embodiments of the disclosure may estimate a 3D body texture from a monocular image, which plays a key role in 3D body reconstruction based on the monocular image, and is widely applied in the aspects of virtual and augmented reality, film industry, games, and biometrics.

[ 00170] In relevant solutions, the 3D body texture (i.e., a UV texture map) may be predicted from an input image (corresponding to an image to be transformed in the above-mentioned embodiments) through a deep CNN. Although the relevant solutions may obtain certain prediction results, the inventor found in researches that there is an inherent disadvantage in its network architecture: a convolutional layer is designed as a local operation, which is difficult to pay attention to vital global information during processing 3D human texture estimation. More specifically, the input and output of the solution do not have a strictly aligned spatial correspondence, and may even have completely different shapes. As shown in FIG. 10, the input of the solution is an input image in FIG. 10 and the output is a texture image in FIG. 10. It can be seen that the input and output of the solution are not only different in image sizes, but also different in the relative position, in the input image, of each body part in the input image and the

relative position, in the texture image, of each body part in the texture image are also different. Therefore, the disclosure introduces global information during extracting texture data, so as to achieve a better 3D body texture reconstruction effect.

[ 00171] In some embodiments, the embodiments of the disclosure provide a transforming module. The transforming module is configured to perform 3D human texture estimation on the basis of a single image. The core of the transforming module is an attention module, which involves three input features: a query feature map, a key feature map, and a value feature map.

[ 00172] In the embodiments of the disclosure, the size of the query feature map is the same as that of an output UV texture map. For example, when the size of the query feature map is  $256 \times 256 \times Q$ , the size of the corresponding UV texture map is  $256 \times 256 \times T$ , i.e., the numbers of the pixel points included in the query feature map and the UV texture map are the same. Each pixel point of the query feature map has a correspondence with a vertex in the 3D body grid. The value feature map includes all source pixels in the image to be transformed. The key feature map includes semantic categories of various pixel points in the image to be transformed. For each pixel point, if the semantic category of the pixel point is a header, a mapping relationship of the pixel point from the image space to the UV space is determined, i.e., corresponds to a header region in the output UV texture map. The key feature map may be a 2D segmentation image obtained by segmenting the image to be transformed on the basis of a semantic segmentation model.

[ 00173] To understand the relationship among the three input features more intuitively, please refer to FIG. 1B, which shows a basic working process of the transforming module. Any pixel (for example, a single pixel that is marked as a star in FIG. 1B) in a query feature is used for correlating a key feature first, so as to generate an attention map. A corresponding pixel in the target texture image (UV map) is generated by effectively aggregating through weighted averaging on the basis of the attention map in combination with the source information in the value feature. On the basis of the transforming module, the global information of the input image can be utilized without losing details. It is to be noted that, for ease of understanding, the implementation mode provided by FIG. 1B is simplified.

[ 00174] FIG. 11 shows a schematic diagram of a model architecture of a texture generation model. It can be seen that the input of the texture generation model includes a query feature B1, a key feature B2, and a value feature B3, and the output is a target texture image B4. The texture generation model includes a feature extraction module B5, a feature transforming module B6, and a feature fusion module B7.

[ 00175] In some embodiments, the query feature B1 is a code feature of an output UV space. Each of the pixel points to be queried in the query feature B1 corresponds to a vertex in a 3D body grid. Therefore, in the disclosure, 3D coordinates of each vertex of a standard body grid from a Skinned Multi-Person Linear (SMPL) Model (a texture-based multi-person linear model) is taken as a color code. A color value of each pixel in the query feature B1 may be obtained in the following manners. First, the 3D coordinates of each vertex of the standard body grid are mapped to the UV space, and then the color value of each pixel in the query feature B1 is obtained by interpolating the coordinates. It is to be noted that the query feature B1 is determined in advance before network training, and the same query feature B1 is used for all input images, i.e., the same color code of the UV space is used.

[ 00176] In some embodiments, the key feature B2 is used for correlating the query feature B1 and generating a corresponding attention map, which is important to connect an image space and a UV space. In the embodiments of the disclosure, the 2D segmentation image of the input image (the key feature map corresponding to the above-mentioned embodiments) may be taken as a key feature for representing a mapping relationship between the input image and the UV texture map. It is to be noted that the above-mentioned 2D segmentation image may be obtained through a semantic segmentation model in relevant solutions, which is the same as that the solution of performing semantic segmentation on the image to be transformed through the two-dimensional semantic segmentation model in the above-mentioned embodiments.

[ 00177] In some embodiments, the key feature B2 also takes the input image as an additional channel in addition to including the semantic feature corresponding to each pixel point in the image space, so as to provide a context feature. For example, when the size of the input image is  $h \times w$ , a 2D segmentation image corresponding to the input

image may be obtained as  $h*w*Y$  through the semantic segmentation model, and is taken as a key feature of the input image. In other embodiments, the input image may also be added to the key feature to obtain the key feature  $h*w*(Y+M)$ .  $M$  is an image feature of the input image at each pixel point, which may include a position feature (coordinates), and may also include a pixel feature.

[ 00178] In some embodiments, the value feature  $B3$  is used for representing source information indexed by the key feature, and is combined with the attention map and aggregated into the UV space to generate an output. The value feature  $B3$  may include the following situations.

[ 00179] (1) When the above-mentioned texture generation model is directly configured to outputting a Red Green Blue (RGB) UV texture map, the value feature  $B3$  may include an RGB feature of the input image.

[ 00180] (2) When the model generates a texture flow feature first and then generates a UV texture map by sampling the input image and a predicted flow, the value feature  $B3$  may input a flow field of the input image, i.e., 2D coordinates of each pixel.

[ 00181] In some embodiments, the value feature  $B3$  may include an RGB feature and a texture flow feature simultaneously, i.e., a fusion feature of the RGB feature and the texture flow feature are taken as the value feature  $B3$ . Correspondingly, the texture generation module may generate a predicted RGB feature and a texture flow feature simultaneously.

[ 00182] The transforming module cannot directly use original forms of the query feature, the key feature, and the value feature, and cannot directly use the original forms of the query feature, the key feature, and the value feature for a comparing process in the attention module. For example, if the original form of the query feature is  $vu \times D1$ , and the original form of the key feature is  $hw \times D2$ , the attention module cannot complete an operating process of the above-mentioned query feature and the key feature when the feature dimension  $D1$  of the query feature is different from the feature dimension  $D2$  of the key feature (i.e.,  $QK^T$ ), so the query feature and the key feature should have the same feature dimension. Therefore, in the embodiments of the disclosure, the above-mentioned

query feature, the key feature, and the value feature are input into the feature extraction module B5 and transformed to the feature space, so as to obtain  $Q \in \mathbb{R}^{vu \times d}$ ,  $K \in \mathbb{R}^{hw \times d}$ ,  $V \in \mathbb{R}^{hw \times c}$ . Then, the obtained feature is sent to the feature transforming module B6. The feature transforming module B6 generates an output feature  $O \in \mathbb{R}^{vu \times d}$ . The  $(v, u)$  is the height and the width of an output UV texture map, the  $(h, w)$  are the height and the width of the input image, and  $d, c$  are respectively the feature dimensions of the  $K$  and  $V$ .

[ 00183] It is to be noted that the solution that the query feature is input into the feature extraction module B5 and transformed to the feature space is only used to indicate the input feature of the feature transforming module B6 at different scales, and is only required in a training stage. In a model deployment stage, the solution that the query feature is input into the feature extraction module B5 and transformed to the feature space may be deleted by calculating a feature code in advance.

[ 00184] In some embodiments, the feature transforming module is a core module of the texture generation model. Image features may be effectively distributed to proper positions of the UV texture map through the feature transforming module, and global information exchange between an input space and an output space may be realized, which may be specifically written as Formula (2-1):

$$O = f_{\text{res-MLP}}(f_{\text{Attn}}(Q + E_Q, K + E_K, V)) \quad \text{Formula (2-1);}$$

where  $f_{\text{Attn}}$  is a multi-head attention module, and  $f_{\text{res-MLP}}$  is a two-layer MLP. There is a residual connection between its input and output.  $E_Q$  and  $E_K$  are respectively sinusoidal position codes of the query feature and the key feature.

[ 00185] In some embodiments, the multi-head attention module  $f_{\text{Attn}}$  is based on a single-head attention module, as shown in Formula (2-2):

$$f_{\text{Attn-S}}(\tilde{Q}, \tilde{K}, \tilde{V}) = \text{softmax}\left(\frac{\tilde{Q}\tilde{K}^T}{\sqrt{d}}\right)\tilde{V} \quad \text{Formula (2-2)}$$

where  $\tilde{Q} \in \mathbb{R}^{\tilde{v}\tilde{u} \times \tilde{d}}$ ,  $\tilde{K} \in \mathbb{R}^{\tilde{h}\tilde{w} \times \tilde{d}}$ , and  $\tilde{V} \in \mathbb{R}^{\tilde{h}\tilde{w} \times \tilde{c}}$ .

[ 00186] An important problem of a common attention formula is high memory

complexity ( $\mathcal{O}(\tilde{v}\tilde{u}\tilde{h}\tilde{w})$ ), which makes a common gpu unable to train. In order to solve the problem, the embodiments of the disclosure use a low rank strategy. The low rank strategy deletes a softmax function from Formula (2-2), and the attention is approximated through pure matrix multiplication, which allows to perform more effective calculation by operating the order of matrix multiplication, i.e., an output result of the attention mechanism is obtained by calculating  $\tilde{Q}\tilde{V}$  first, and then combining  $\tilde{K}^T$ . In this way, memory occupation may be greatly reduced to ( $\mathcal{O}(\max(\tilde{v}\tilde{u}, \tilde{h}\tilde{w}) \cdot (\max(\tilde{d}, \tilde{c}))$ ), which makes the training be more feasible.

[ 00187] In some embodiments, the texture generation model adopts a solution of multi-scale feature fusion. That is, an output feature is calculated through a three-level feature pyramid instead of applying the feature transforming module B6 on a single scale. Taking FIG. 11 that the feature transforming module B6 is applied on three scales as an example, the texture generation model may further include a feature transforming module B61 corresponding to a first scale, a feature transforming module B62 corresponding to a second scale, and a feature transforming module B63 corresponding to a third scale. The third scale is greater than the second scale, and the second scale is greater than the first scale. Correspondingly, for the scales (i.e., the second scale and the third scale) of first two relatively large feature spaces, the embodiments of the disclosure use the above-mentioned low rank strategy to reduce memory overhead. For the scale (i.e., the first space) of a third relatively small feature space, the embodiments of the disclosure use a common softmax-based attention module. Then, obtained multi-scale output features  $O_i$ ,  $i=1, 2, 3$  are fused through a feature fusion module B7. The multi-scale feature fusion solution can make the feature transforming module B6 use context information better, and predict a high-quality body texture in an invisible region (i.e., an occluded texture region, or a textured region with a complex structure).

[ 00188] In some embodiments, there are two possible options for the output of the texture generation model: (1) RGB texture  $T_{RGB}$  (corresponding to the pixel feature map in the above-mentioned embodiments) is generated directly. (2) A texture flow  $F$  (corresponding to the texture flow feature map in the above-mentioned embodiments) that can be used for generating a final texture is predicted by sampling from the input image  $I$ . Referring to FIG. 12, these two strategies have their own advantages and

limitations: visually satisfactory 3D body texture may be reconstructed through direct synthesis  $T_{RGB}$ , but details are often lost (an RGB rendering result in FIG. 12). On another aspect, the texture flow can retain detail information in the input image, but severe artifacts may exist (the texture flow rendering result in FIG. 12).

[ 00189] In order to solve the above-mentioned problems, the embodiments of the disclosure adopt a mask-fusion solution to combine the advantages of the above-mentioned RGB texture map and the texture flow. Therefore, the output of the feature transforming module in the embodiments of the disclosure not only includes the RGB texture map and the texture flow  $F$ , but also includes a mask feature (corresponding to a confidence feature map in the above-mentioned embodiments). As shown in FIG. 11, the feature fusion module B7 may not only adopt a plurality of downsampling processing and a plurality of upsampling processing to fuse the outputs of a plurality of scales output by the feature transforming module, and perform “ $\mathcal{F}$ ” operation on the RGB texture map, the texture flow, and the mask feature obtained by fusion to obtain a final texture map (corresponding to the target texture map in the above-mentioned embodiments.) The above-mentioned feature fusion module B7 may execute the “ $\mathcal{F}$ ” operation process as Formula (2-3):

$$T = M \odot f_{\text{sample}}(F, I) + (1 - M) \odot T_{RGB} \quad \text{Formula (2-3)}$$

where  $f_{\text{sample}}$  is a bilinear sampling function, the  $f_{\text{sample}}$  is used for sampling from the input image  $I$  on the basis of the texture flow  $F$ , so as to obtain a corresponding first texture map.  $\odot$  represents multiplying element by element, and the final texture  $T$  is calculated by fusing  $T_{RGB}$  and  $f_{\text{sample}}(F, I)$  in a weighted summation manner. As shown in the mask-fusion rendering result in FIG. 12, the mask-fusion method can remarkably reduce visual artifacts, and meanwhile, retain fine details.

[ 00190] In addition, by observing a learned mask feature in FIG. 12, the inventor found that the texture generation model tends to use a sampled texture sample  $(F, I)$  to serve as a visible region (for example, body chest), and use a synthesis texture  $T_{RGB}$  to serve as an invisible region, for example, human back or other regions with complex structures (for example, face and hand).

[ 00191] For the texture generation model, the embodiments of the disclosure provide a plurality of the following loss functions, so as to improve the texture generation effect of the texture generation model.

[ 00192] In some embodiments, the embodiments of the disclosure may render on the basis of the obtained target texture image to obtain a corresponding rendering texture image  $f_r(T, D)$ , herein,  $f_r$  is a differentiable rendering function, and  $D$  represents a 3D body grid and camera parameter predicted by a relevant algorithm. That is to say, the target texture image is rendered to a 3D body model corresponding to a body posture in the input image, a corresponding projection image is generated through a virtual camera, and the projection image is taken as the above-mentioned rendering texture image. Later, the texture generation model may be trained by improving the similarity between the rendering image  $f_r(T)$  and the input image  $I$ . Please note that  $f_r(T, D)$  will be replaced with  $f_r(T)$  in the embodiments below for sake of simplicity.

[ 00193] Therefore, in order to enhance the similarity between the  $f_r(T)$  and  $I$ , the embodiments of the disclosure provide a first loss function ( $l_{\text{ReID}}$ ), as shown in Formula (2-4):

$$l_{\text{ReID}} = \sum_v \|\bar{\phi}_v(f_r(T)) - \bar{\phi}_v(I)\|_2^2 \quad \text{Formula (2-4)}$$

where  $\phi_v$  represents the  $v$ th layer of the Person re-identification (ReID) network.  $\bar{\phi} = \phi / \|\phi\|_2$  represents L2-normalization.  $l_{\text{ReID}}$  corresponds to the first loss value in the above-mentioned embodiments.

[ 00194] Through research, the inventor found that a first loss function may restore texture information in the input image to a certain degree, but a reconstructed texture usually has an obvious color difference from the input image, as shown in a ReID rendering result in FIG. 13, which is caused by a normalization operation in Formula (2-4). A method for solving the problem is to a de-normalization operation, and a feature that is not normalized is used as a first loss function. However, this strategy usually leads to serious artifacts in the reconstructed body texture, and as shown a de-normalized ReID rendering result in FIG. 13, there are unnatural textures on the reconstructed human face

and arms. Through certain experiments, the inventor found that the above-mentioned problem is caused by that the 3D body grid predicted by relevant algorithms is not always accurate. The 3D body grid is not aligned with the input image very well. Therefore, an adverse effect may be produced on a training process by only taking the similarity between a rendered body and the input image as a loss value of model training.

[ 00195] In order to solve the problem, the embodiments of the disclosure further provide a second loss function, which is based on the observation that a 2D segmentation result is usually more accurate than a 3D segmentation result and is aligned with the input image better. Specifically, the embodiments of the disclosure determines the similarity between the rendered body and the input image for each body part, which can solve the problem of misalignment caused by inaccurate prediction of the 3D grid.

[ 00196] However, the regions of the same body part in different segmentation images usually have different sizes and shapes. The segmentation image obtained by rendering on the basis of the 3D grid and the segmentation image obtained on the basis of a 2D input image also have different sizes and shapes at the same body part, so the loss value calculation method of the above-mentioned Formula (2-4) cannot be adopted. According to the embodiments of the disclosure, the second loss value is determined through Gram matrix. The Gram matrix does not need the same size and shape during calculating. Different from a solution for calculating the similarity of the Gram matrix on the basis of an overall image, the embodiments of the disclosure obtain the second loss function ( $l_{Style}$ ) by calculating the similarity of the Gram matrix corresponding to each body part through each body part, as shown in Formula (2-5):

$$l_{Style} = \sum_p \left\| G \left( M_p \odot \phi_1(f_r(T)) \right) - G \left( M'_p \odot \phi_1(I) \right) \right\|_2^2 \quad \text{Formula (2-5)}$$

where  $M$  and  $M'$  are respectively body part segmentation results from the 3D grid or a 2D body analytical model. Here,  $p$  represents a  $p$ th body part, and  $G$  is the Gram matrix. The second loss value ( $l_{Style}$ ) is determined by using the feature (i.e., " $\phi_1$ ") of the first layer of the ReID network, so that the second loss function may pay better attention to a low level color of the reconstructed texture.

[ 00197] As shown in a ReID+Style rendering result in FIG. 13, the first loss value ( $l_{\text{ReID}}$ ) obtained on the basis of the first loss function and the second loss value ( $l_{\text{Style}}$ ) obtained on the basis of the second loss function may realize a color appearance that is closer to the input image without introducing unpleasant artifacts.

[ 00198] In some embodiments, 3D textures in most regions of a body can be reconstructed very well on the basis of above-mentioned solution of the first loss value and the second loss value, but it is still challenging to reconstruct the textures of the body parts with complex structures (for example, faces). In relevant technologies, the face texture may be obtained in a manner of encouraging facial textures to approach the average facial texture of a human dataset. However, this usually causes that a facial color is inconsistent with other skin, thereby causing an unnatural situation. In order to solve this problems, the embodiments of the disclosure provide a loss of a facial structure  $l_{\text{face}}$  to serve as a third loss function by analyzing and improving the original Structural Similarity (SSIM). The generation of a reasonable facial texture may be promote effectively through the third loss function, and meanwhile, the color of the facial texture and the color of an input body are kept the same.

[ 00199] On the basis of the above-mentioned embodiments, the global loss function of the above-mentioned texture generation model include the first loss function, the second loss function, and the third loss function simultaneously. In some embodiments, the global loss function may be shown as Formula (2-6):

$$l = w_1 * l_{\text{ReID}} + w_2 * l_{\text{Style}} + w_3 * l_{\text{face}} \quad \text{Formula (2-6)}$$

where  $w_1$ ,  $w_2$  and  $w_3$  are hyper-parameters. In some specific embodiments,  $w_1 = 5000$ ,  $w_2 = 0.4$ , and  $w_3 = 0.01$ .

[ 00200] FIG. 14 is a schematic diagram of a composition structure of a texture generation apparatus provided by the embodiments of the disclosure. As shown in FIG. 14, the texture generation apparatus 1400 includes an acquisition module 1401 and a generation module 1402.

[ 00201] The acquisition module 1401 is configured to acquire a query feature map, a

key feature map, and a value feature map of an image to be transformed. The query feature map includes a plurality of pixel points to be queried in a texture space. The key feature map is used for representing a semantic feature of the image to be transformed. The value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried. The attention feature map is determined by the query feature map and the key feature map.

[ 00202]       The generation module 1402 is configured to generate a target texture feature map from the query feature map, the key feature map, and the value feature map on the basis of an attention mechanism. The target texture feature map includes a texture parameter corresponding to each of the pixel points to be queried.

[ 00203]       In some embodiments, the generation module 1402 is further configured to perform the following operations.

[ 00204]       Feature extraction processing at different scales is performed on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales. The input feature map includes the query feature map, the key feature map, and the value feature map.

[ 00205]       An input feature corresponding to each of the scales is input into a corresponding transforming module to obtain an output feature corresponding to each of the scales.

[ 00206]       The output features corresponding to all of the scales are fused to obtain the target texture feature map.

[ 00207]       In some embodiments, the generation module 1402 is further configured to perform the following operations.

[ 00208]       Downsampling processing is performed on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale.

[ 00209]       Upsampling processing is performed on the input feature corresponding to the

preset scale for a plurality of times to obtain input features corresponding to the plurality of scales. The number of times of the downsampling processing is the same as that of the upsampling processing.

[ 00210] In some embodiments, the generation module 1402 is further configured to perform the following operations.

[ 00211] For each scale, the query feature map and a position code of the query feature map are fused to obtain a first input feature.

[ 00212] The key feature map and a position code of the key feature map are fused to obtain a second input feature.

[ 00213] An intermediate feature map is obtained on the basis of the first input feature, the second input feature, and the value feature map.

[ 00214] The output feature corresponding to the scale is determined on the basis of the intermediate feature map and a residual network.

[ 00215] In some embodiments, the generation module 1402 is further configured to perform the following operations.

[ 00216] The output features corresponding to all of the scales are fused to obtain a multi-scale output feature.

[ 00217] The target texture feature map is determined on the basis of the multi-scale output feature.

[ 00218] In some embodiments, the generation module 1402 is further configured to perform the following operations.

[ 00219] Downsampling fusion processing is performed on the output feature corresponding to each of the scales in turn according to an order from the largest scale to the smallest scale of the plurality of scales to obtain a downsampling fusion feature corresponding to each of the scales. The downsampling fusion processing is used for fusing the output feature corresponding to the scale and a downsampling fusion feature

obtained from a last downsampling fusion processing to obtain a downsampling intermediate feature, and performing downsampling processing on the downsampling intermediate feature to obtain a downsampling fusion feature corresponding to the scale.

[ 00220] Upsampling fusion processing is performed on the output features corresponding to each of the scales in turn according to an order from the smallest scale to the largest scale of the plurality of scales to obtain an upsampling fusion feature corresponding to each of the scales. The upsampling fusion processing is used for fusing the output feature corresponding to the scale and an upsampling fusion feature obtained from a last downsampling fusion processing to obtain an upsampling intermediate feature, and performing upsampling processing on the upsampling intermediate feature to obtain an upsampling fusion feature corresponding to the scale.

[ 00221] The upsampling fusion processing of the smallest scale is used for performing upsampling processing on the downsampling fusion feature corresponding to the smallest scale, and fusing the output feature corresponding to the smallest scale, to obtain an upsampling fusion feature corresponding to the smallest scale. The upsampling fusion feature corresponding to the largest scale is a multi-scale output feature.

[ 00222] In some embodiments, in the case where the multi-scale output feature includes a pixel feature map, the generation module 1402 is further configured to perform the following operation.

[ 00223] The pixel feature map is taken as the target texture feature map.

[ 00224] In some embodiments, in the case where the multi-scale output feature includes a texture flow feature map, the generation module 1402 is further configured to perform the following operation.

[ 00225] The target texture feature map is generated on the basis of the texture flow feature map and the image to be transformed.

[ 00226] In some embodiments, in the case where the multi-scale output feature includes the pixel feature map, the texture flow feature map, and a confidence feature map, the value feature map includes the original pixel feature and the pixel distribution

feature of the input image. The generation module 1402 is further configured to perform the following operations.

[ 00227] Confidence information of each of the pixel points to be queried in the texture space is determined on the basis of the confidence feature map. The confidence information includes a first confidence of the pixel point to be queried corresponding to the pixel feature map and a second confidence of the pixel point to be queried corresponding to the texture flow feature map.

[ 00228] The pixel feature map and the texture flow feature map are fused on the basis of the first confidence and the second confidence of each of the pixel points to be queried to obtain the target texture feature map.

[ 00229] In some embodiments, the generation module 1402 is further configured to perform the following operations.

[ 00230] A first texture parameter corresponding to each of the pixel points to be queried is determined on the basis of the pixel feature map.

[ 00231] A second texture parameter corresponding to each of the pixel points to be queried is determined on the basis of the texture flow feature map and the image to be transformed.

[ 00232] For each of the pixel points to be queried, weighted fusion is performed on the first texture parameter and the second texture parameter on the basis of the first confidence and the second confidence respectively to obtain a texture parameter corresponding to the pixel point to be queried.

[ 00233] In some embodiments, the texture generation method provided by the above-mentioned embodiments is implemented by a trained texture generation model. The texture generation apparatus further includes a training module. The training module is configured to perform the following operations.

[ 00234] A sample image is acquired. The sample image is input into a texture generation model to be trained to obtain a predicted texture image corresponding to the

sample image.

- [ 00235]        The predicted texture image corresponding to the sample image is rendered to obtain a predicted rendered image corresponding to the sample image.
- [ 00236]        A loss value of the texture generation model to be trained is determined on the basis of the sample image and the predicted rendered image.
- [ 00237]        A parameter of the texture generation model to be trained according to the loss value is adjusted to obtain a trained texture generation model.
- [ 00238]        In some embodiments, the loss value of the texture generation model to be trained includes a first loss value.
- [ 00239]        The training module is further configured to perform the following operations. A first character feature in the sample image and a second character feature in the predicted rendered image are respectively extracted on the basis of a person ReID model. The first loss value is determined on the basis of a feature distance between the first character feature and the second character feature. A parameter of the texture generation model to be trained is adjusted on the basis of at least the first loss value to obtain the trained texture generation model.
- [ 00240]        In some embodiments, the loss value of the texture generation model to be trained includes a second loss value. The training model is further configured to perform the following operations. A first character feature in the sample image and a second character feature in the predicted rendered image are respectively extracted on the basis of a person ReID model. A first segmentation result corresponding to the sample image is acquired. The first segmentation result is used for representing first distribution information of each body part in the sample image. A second segmentation result corresponding to the predicted rendered image is acquired. The second segmentation result is used for representing second distribution information of each body part in the predicted rendered image. For each body part, a part loss value corresponding to the body part is determined on the basis of a feature distance between a first part feature and a second part feature of the body part. The first part feature is determined on the basis of the first character feature and first part information of the body part. The second part

feature is determined on the basis of the second character feature and second part information of the body part. The second loss value is determined on the basis of the part loss value corresponding to each body part. The parameter of the texture generation model to be trained is adjusted on the basis of at least the second loss value to obtain the trained texture generation model.

[ 00241] In some embodiments, the loss value of the texture generation model to be trained includes a third loss value. The training module is further configured to perform the following operations. A part texture image corresponding to a target body part in the predicted texture image is acquired. A standard texture image corresponding to the target body part is acquired. The standard texture image is a fusion image of part texture images of different characters regarding the target body part. The third loss value is determined on the basis of a structural similarity index between the part texture image and the standard texture image. The parameter of the texture generation model to be trained is adjusted at the basis of least on the third loss value to obtain the trained texture generation model.

[ 00242] The above descriptions about the apparatus embodiments are similar to descriptions about the method embodiments and beneficial effects similar to those of the method embodiments are achieved. Technical details undisclosed in the apparatus embodiments of the disclosure may be understood with reference to the descriptions about the method embodiments of the disclosure.

[ 00243] It is to be noted that, in the embodiments of the disclosure, when implemented in form of software function module and sold or used as an independent product, the warning method may also be stored in a computer-readable storage medium. Based on such an understanding, the technical solutions of the embodiments of the disclosure substantially or parts making contributions to the related art may be embodied in form of a software product. The computer software product is stored in a storage medium, including a plurality of instructions configured to enable a device to execute all or part of the method in each embodiment of the disclosure. The above-mentioned storage medium includes: various media capable of storing program codes, such as a USB flash drive, a mobile hard disk, a Read-Only Memory (ROM), a magnetic disk, or a compact disk.

Therefore, the embodiments of the disclosure are not limited to any target hardware and software combination.

[ 00244] FIG. 15 is a schematic diagram of a hardware entity of a texture generation device provided by the embodiments of the disclosure. As shown in FIG. 15, the hardware entity of the texture generation device 1500 includes: a processor 1501 and a memory 1502. The memory 1502 stores a computer program capable of running in the processor 1501. The processor 1501 executes the program to implement the steps in the method of any above-mentioned embodiment. In some implementation modes, the device 1500 that collects and pays tokens on a game table may be the detection device described in any above-mentioned embodiment.

[ 00245] The memory 1502 stores the computer program capable of running in the processor 1501. The memory 1502 is configured to store an instruction and application executable for the processor 1501, may also cache data (for example, image data, audio data, voice communication data, and video communication data) to be processed or having been processed by the processor 1501 and each module in the texture generation device 1500, and may be implemented through a flash or a Random Access Memory (RAM).

[ 00246] The processor 1501 executes the program to implement the steps of any above-mentioned texture generation method. The processor 1501 usually controls overall operations of the texture generation device 1500.

[ 00247] The embodiments of the disclosure provide a computer storage medium, which stores one or more programs. The one or more programs may be executed by one or more processors to implement the steps of the texture generation method in any above-mentioned embodiment.

[ 00248] It is to be pointed out here that the above descriptions about the storage medium and device embodiments are similar to the descriptions about the method embodiment and beneficial effects similar to those of the method embodiment are achieved. Technical details undisclosed in the storage medium and device embodiments of the disclosure are understood with reference to the descriptions about the method

embodiment of the disclosure.

[ 00249]       The processor may be at least one of an Application Specific Integrated Circuit (ASIC), a Digital Signal Processor (DSP), a Digital Signal Processing Device (DSPD), a Programmable Logic Device (PLD), a Field Programmable Gate Array (FPGA), a Central Processing unit (CPU), a controller, a microcontroller, or a microprocessor. It can be understood that other electronic devices may also be configured to realize functions of the processor, and no specific limits are made in the embodiments of the disclosure.

[ 00250]       The computer storage medium/memory may be a memory such as an ROM, a Programmable Read-Only Memory (PROM), an Erasable Programmable Read-Only Memory (EPROM), an Electrically Erasable Programmable Read-Only Memory (EEPROM), a Ferromagnetic Random Access Memory (FRAM), a flash memory, a magnetic surface memory, an optical disk, or a Compact Disc Read-Only Memory (CD-ROM), or may be any terminal including one or any combination of the above-mentioned memories, such as a mobile phone, a computer, a tablet device, and a personal digital assistant.

[ 00251]       It is to be understood that "one embodiment" or "an embodiment" or "the embodiment of the disclosure" or "the above-mentioned embodiment" or "some embodiments" mentioned in the whole specification means that target features, structures or characteristics related to the embodiment are included in at least one embodiment of the disclosure. Therefore, "in one embodiment" or "in an embodiment" or "the embodiment of the disclosure" or "the above-mentioned embodiment" or "some embodiments" appearing everywhere in the whole specification does not always refer to the same embodiment. In addition, these target 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 disclosure, a magnitude of a 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 disclosure. The sequence numbers of the embodiments of the disclosure are adopted not to represent superiority-inferiority of the embodiments but only for description.

[ 00252] If not specified, when the warning device executes any step in the embodiments of the disclosure, the processor of the warning device executes the step. Unless otherwise specified, the sequence of execution of the following steps by the warning device is not limited in the embodiments of the disclosure. In addition, the same method or different methods may be used to process data in different embodiments. It is also to be noted that any step in the embodiments of the disclosure may be executed independently by the warning device, namely the warning device may execute any step in the above-mentioned embodiments independent of execution of the other steps.

[ 00253] In some embodiments provided by the disclosure, it is to be understood that the disclosed device and method may be implemented in other manners. 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, a plurality of 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.

[ 00254] 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 a plurality of 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.

[ 00255] In addition, each function unit in each embodiment of the disclosure 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.

[ 00256] The methods disclosed in some method embodiments provided in the disclosure may be freely combined without conflicts to obtain new method embodiments.

[ 00257]       The characteristics disclosed in several product embodiments provided in the disclosure may be freely combined without conflicts to obtain new product embodiments.

[ 00258]       The characteristics disclosed in several method or device embodiments provided in the disclosure may be freely combined without conflicts to obtain new method embodiments or device embodiments.

[ 00259]       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 a compact disk.

[ 00260]       Or, when implemented in form of software function module and sold or used as an independent product, the integrated unit of the disclosure may also be stored in a computer-readable storage medium. Based on such an understanding, the technical solutions of the embodiments of the disclosure substantially or parts making contributions to the related art may be embodied in form of a software product. The computer software product is stored in a storage medium, including a plurality of instructions configured to cause a computer device (which may be a personal computer, a warning device, a network device or the like) to execute all or part of the method in each embodiment of the disclosure. The previously described storage medium includes: various media capable of storing program codes, such as a mobile storage device, the ROM, a magnetic disk, or a compact disk.

[ 00261]       In the embodiments of the disclosure, the descriptions about the same steps and the same contents in different embodiments may refer to those in the other embodiments. In the embodiments of the disclosure, term "and" does not cause influence on the sequence of the steps.

[ 00262]       The above-mentioned descriptions are only implementation modes of the disclosure, but are not intended to limit the protection scope of the disclosure. Any variation or replacement readily figured out by a person skilled in the art within the

technical scope disclosed in this disclosure shall fall within the protection scope of this disclosure. Therefore, the protection scope of this disclosure shall be subject to the protection scope of the claims.

## CLAIMS

### 1. A texture generation method, comprising:

acquiring a query feature map, a key feature map, and a value feature map of an image to be transformed, wherein the query feature map comprises a plurality of pixel points to be queried in a texture space; the key feature map is used for representing a semantic feature of the image to be transformed; the value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried; the attention feature map is determined by the query feature map and the key feature map; and

generating a target texture feature map from the query feature map, the key feature map, and the value feature map based on an attention mechanism, wherein the target texture feature map comprises the texture parameter corresponding to each of the pixel points to be queried;

wherein the generating a target texture feature map from the query feature map, the key feature map, and the value feature map based on an attention mechanism comprises:

performing feature extraction processing at different scales on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales, wherein the input feature map comprises the query feature map, the key feature map, and the value feature map;

inputting an input feature corresponding to each of the scales into a corresponding transforming module to obtain an output feature corresponding to each of the scales; and

fusing the output features corresponding to all of the scales to obtain the target texture feature map;

wherein the performing feature extraction processing at different scales on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales comprises:

performing downsampling processing on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale; and

performing upsampling processing on the input feature corresponding to the preset scale for a plurality of times to obtain input features corresponding to the plurality of scales, wherein a number of times of the downsampling processing is the same as a number of times of the upsampling processing.

2. The method of claim 1, wherein the inputting an input feature corresponding to each of the scales into a corresponding transforming module to obtain an output feature corresponding to each of the scales comprises:

for each scale, fusing the query feature map and a position code of the query feature map to obtain a first input feature;

fusing the key feature map and a position code of the key feature map to obtain a second input feature;

obtaining an intermediate feature map based on the first input feature, the second input feature, and the value feature map; and

determining the output feature corresponding to the scale based on the intermediate feature map and a residual network.

3. The method of claim 1, wherein the fusing the output features corresponding to all of the scales to obtain the target texture feature map comprises:

fusing the output features corresponding to all of the scales to obtain a multi-scale output feature; and

determining the target texture feature map based on the multi-scale output feature.

4. The method of claim 3, wherein the fusing the output features corresponding to all of the scales to obtain a multi-scale output feature comprises:

performing downsampling fusion processing on the output feature corresponding to each of the scales in turn according to an order from a largest scale to a smallest scale of the plurality of scales to obtain a downsampling fusion feature corresponding to each of the scales, wherein the downsampling fusion processing is used for fusing the output feature corresponding to the scale and a downsampling fusion feature obtained from a last downsampling fusion processing to obtain a downsampling intermediate feature, and performing downsampling processing on the downsampling intermediate feature to obtain a downsampling fusion feature corresponding to the scale; and

performing upsampling fusion processing on the output feature corresponding to each of the scales in turn according to an order from the smallest scale to the largest scale of the plurality of scales to obtain an upsampling fusion feature corresponding to each of the scales, wherein the upsampling fusion processing is used for fusing the output feature corresponding to the scale and an upsampling fusion feature obtained from a last upsampling fusion processing to obtain an upsampling intermediate feature, and performing upsampling processing on the upsampling intermediate feature to obtain an upsampling fusion feature corresponding to the scale,

wherein the upsampling fusion processing of the smallest scale is used for performing upsampling processing on a downsampling fusion feature corresponding to the smallest scale, and fusing the output feature corresponding to the smallest scale, to obtain an upsampling fusion feature corresponding to the smallest scale; and an upsampling fusion feature corresponding to the largest scale is a multi-scale output feature.

5. The method of claim 3, wherein in a case where the multi-scale output feature comprises a pixel feature map, the value feature map comprises an original pixel feature of the image to be transformed; the determining the target texture feature map based on the multi-scale output feature comprises:

taking the pixel feature map as the target texture feature map.

6. The method of claim 3, wherein in a case where the multi-scale output feature comprises a texture flow feature map, the value feature map comprises a pixel distribution feature of the image to be transformed; the determining the target texture feature map based on the multi-scale output feature comprises:

generating the target texture feature map based on the texture flow feature map and the image to be transformed.

7. The method of claim 3, wherein in a case where the multi-scale output feature comprises a pixel feature map, a texture flow feature map and a confidence feature map, the value feature map comprises an original pixel feature and a pixel distribution feature of the image to be transformed; the determining the target texture feature map based on the multi-scale output feature comprises:

determining confidence information of each of the pixel points to be queried in the texture space based on the confidence feature map, wherein the confidence information comprises a first confidence of the pixel point to be queried corresponding to the pixel feature map and a second confidence of the pixel point to be queried corresponding to the texture flow feature map; and

fusing the pixel feature map and the texture flow feature map based on the first confidence and the second confidence of each of the pixel points to be queried, to obtain the target texture feature map.

8. The method of claim 7, wherein the fusing the pixel feature map and the texture flow feature map based on the first confidence and the second confidence of each of the pixel points to be queried, to obtain the target texture feature map comprises:

determining a first texture parameter corresponding to each of the pixel points to be queried based on the pixel feature map;

determining a second texture parameter corresponding to each of the pixel points to be queried based on the texture flow feature map and the image to be transformed; and

for each of the pixel points to be queried, performing weighted fusion on the first texture parameter and the second texture parameter based on the first confidence and the second confidence respectively to obtain a texture parameter corresponding to the pixel point to be queried.

9. The method of any one of claims 1 to 8, wherein the method is implemented by a trained texture generation model; a training process of the texture generation model comprises:

acquiring an sample image, and inputting the sample image into a texture generation model to be trained to obtain a predicted texture image corresponding to the sample image;

rendering the predicted texture image corresponding to the sample image to obtain a predicted rendered image corresponding to the sample image;

determining a loss value of the texture generation model to be trained based on the sample image and the predicted rendered image; and

adjusting, according to the loss value, a parameter of the texture generation model to be trained to obtain the trained texture generation model.

10. The method of claim 9, wherein the loss value of the texture generation model to be trained comprises a first loss value;

wherein the determining a loss value of the texture generation model to be trained based on the sample image and the predicted rendered image comprises:

extracting a first character feature in the sample image and a second character feature in the predicted rendered image respectively based on a person re-identification model; and

determining the first loss value based on a feature distance between the first character feature and the second character feature;

wherein the adjusting, according to the loss value, a parameter of the texture generation model to be trained to obtain the trained texture generation model comprises:

adjusting, based on at least the first loss value, the parameter of the texture generation model to be trained to obtain the trained texture generation model.

11. The method of claim 9, wherein the loss value of the texture generation model to be trained comprises a second value;

wherein the determining a loss value of the texture generation model to be trained based on the sample image and the predicted rendered image comprises:

extracting a first character feature in the sample image and a second character feature in the predicted rendered image respectively based on a person re-identification model;

acquiring a first segmentation result corresponding to the sample image, wherein the first segmentation result is used for representing first distribution information of each body part in the sample image;

acquiring a second segmentation result corresponding to the predicted rendered image, wherein the second segmentation result is used for representing second distribution information of each body part in the predicted rendered image;

for each body part, determining a part loss value corresponding to the body part based on a feature distance between a first part feature and a second part feature of the body part, wherein the first part feature is determined based on the first character feature and first part information of the body part, and the second part feature is determined based on the second character feature and second part information of the body part; and

determining the second loss value based on the part loss value corresponding to the each body part;

wherein the adjusting, according to the loss value, a parameter of the texture generation model to be trained to obtain the trained texture generation model comprises:

adjusting, based on at least the second loss value, the parameter of the texture generation model to be trained to obtain the trained texture generation model.

12. The method of claim 9, wherein the loss value of the texture generation model to be trained comprises a third loss value, the method further comprises:

acquiring a part texture image corresponding to a target body part in the predicted texture image;

acquiring a standard texture image corresponding to the target body part, wherein the standard texture image is a fusion image of part texture images of different characters regarding the target body part; and

determining the third loss value based on a structural similarity index between the part texture image and the standard texture image;

wherein the adjusting, according to the loss value, a parameter of the texture generation model to be trained to obtain the trained texture generation model comprises:

adjusting, based on at least the third loss value, the parameter of the texture generation model to be trained to obtain the trained texture generation model.

13. A texture generation apparatus, comprising:

an acquisition module, configured to acquire a query feature map, a key feature map, and a value feature map of an image to be transformed, the query feature map comprising a plurality of pixel points to be queried in a texture space, the key feature map being used for representing a semantic feature of the image to be transformed, the value feature map being used for determining, in combination with an attention feature map of each of the pixel points

to be queried, a texture parameter of each of the pixel points to be queried, and the attention feature map being determined by the query feature map and the key feature map; and

a generation module, configured to generate a target texture feature map from the query feature map, the key feature map, and the value feature map based on an attention mechanism, and the target texture feature map comprising the texture parameter corresponding to each of the pixel points to be queried;

wherein the generation module is further configured to:

perform feature extraction processing at different scales on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales, wherein the input feature map comprises the query feature map, the key feature map, and the value feature map;

input an input feature corresponding to each of the scales into a corresponding transforming module to obtain an output feature corresponding to each of the scales; and

fuse the output features of corresponding to all of the scales to obtain a target texture feature map;

wherein the generation module is configured to:

perform downsampling processing on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale; and

perform upsampling processing on the input feature corresponding to the preset scale for a plurality of times to obtain input features corresponding to the plurality of scales, wherein a number of times of the downsampling processing is the same as a number of times of the upsampling processing.

14. The apparatus of claim 13, further comprising a training module,

wherein the training module is configured to:

acquire a sample image, and input the sample image into a texture generation model to be trained to obtain a predicted texture image corresponding to the sample image;

render the predicted texture image corresponding to the sample image to obtain a predicted rendered image corresponding to the sample image;

determine a loss value of the texture generation model to be trained based on the sample image and the predicted rendered image; and

adjust, according to the loss value, a parameter of the texture generation model to be trained to obtain the trained texture generation model.

15. A texture generation device, comprising:

a memory, configured to store an executable instruction; and

a processor, configured to execute the executable instruction stored in the memory to:

acquire a query feature map, a key feature map, and a value feature map of an image to be transformed, wherein the query feature map comprises a plurality of pixel points to be queried in a texture space; the key feature map is used for representing a semantic feature of the image to be transformed; the value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried; the attention feature map is determined by the query feature map and the key feature map; and

generate a target texture feature map from the query feature map, the key feature map, and the value feature map based on an attention mechanism, wherein the target texture feature map comprises the texture parameter corresponding to each of the pixel points to be queried;

wherein the processor is configured to:

perform feature extraction processing at different scales on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales, wherein

the input feature map comprises the query feature map, the key feature map, and the value feature map;

input an input feature corresponding to each of the scales into a corresponding transforming module to obtain an output feature corresponding to each of the scales; and

fuse the output features corresponding to all of the scales to obtain the target texture feature map;

wherein the processor is configured to:

perform downsampling processing on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale; and

perform upsampling processing on the input feature corresponding to the preset scale for a plurality of times to obtain input features corresponding to the plurality of scales, wherein a number of times of the downsampling processing is the same as a number of times of the upsampling processing.

16. A computer readable storage medium, storing an executable instruction, wherein the executable instruction, when executed by a processor, is configured to:

acquire a query feature map, a key feature map, and a value feature map of an image to be transformed, wherein the query feature map comprises a plurality of pixel points to be queried in a texture space; the key feature map is used for representing a semantic feature of the image to be transformed; the value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried; the attention feature map is determined by the query feature map and the key feature map; and

generate a target texture feature map from the query feature map, the key feature map, and the value feature map based on an attention mechanism, wherein the target texture feature map comprises the texture parameter corresponding to each of the pixel points to be queried;

wherein the executable instruction is configured to:

perform feature extraction processing at different scales on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales, wherein the input feature map comprises the query feature map, the key feature map, and the value feature map;

input an input feature corresponding to each of the scales into a corresponding transforming module to obtain an output feature corresponding to each of the scales; and

fuse the output features corresponding to all of the scales to obtain the target texture feature map;

wherein the executable instruction is configured to:

perform downsampling processing on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale; and

perform upsampling processing on the input feature corresponding to the preset scale for a plurality of times to obtain input features corresponding to the plurality of scales, wherein a number of times of the downsampling processing is the same as a number of times of the upsampling processing.

17. A computer program, comprising a computer program code, wherein when the computer program code is executed by an electronic device, a processor in the electronic device is configured to:

acquire a query feature map, a key feature map, and a value feature map of an image to be transformed, wherein the query feature map comprises a plurality of pixel points to be queried in a texture space; the key feature map is used for representing a semantic feature of the image to be transformed; the value feature map is used for determining, in combination with an attention feature map of each of the pixel points to be queried, a texture parameter of each of the pixel points to be queried; the attention feature map is determined by the query feature map and the key feature map; and

generate a target texture feature map from the query feature map, the key feature map, and the value feature map based on an attention mechanism, wherein the target texture feature map comprises the texture parameter corresponding to each of the pixel points to be queried;

wherein the processor is configured to:

perform feature extraction processing at different scales on an input feature map of the attention mechanism to obtain input features corresponding to a plurality of scales, wherein the input feature map comprises the query feature map, the key feature map, and the value feature map;

input an input feature corresponding to each of the scales into a corresponding transforming module to obtain an output feature corresponding to each of the scales; and

fuse the output features corresponding to all of the scales to obtain the target texture feature map;

wherein the processor is configured to:

perform downsampling processing on the input feature map for a plurality of times to obtain an input feature corresponding to a preset scale; and

perform upsampling processing on the input feature corresponding to the preset scale for a plurality of times to obtain input features corresponding to the plurality of scales, wherein a number of times of the downsampling processing is the same as a number of times of the upsampling processing.