

METHODS FOR IDENTIFYING AN OBJECT SEQUENCE IN AN IMAGE, TRAINING METHODS, APPARATUSES AND DEVICES

TECHNICAL FIELD

[0001] The present disclosure relates to the field of artificial intelligence technology, and in particular, to a method for identifying an object sequence in an image, a training method, an apparatus and a device.

BACKGROUND

[0002] In some scenarios, it is necessary to identify an object sequence formed by stacking a plurality of objects in an image, in order to determine a category and attribute of each of the plurality of objects in the object sequence. However, since the stacked objects in the object sequence are often similar to each other and there are many ways to combine different objects in the stacking process, it is difficult to accurately identify the category of each of the objects in the object sequence, and in the related art, the accuracy of results for identifying the object sequence in the image is often low, which still needs to be improved.

SUMMARY

[0003] The present disclosure provides a method for identifying an object sequence in an image, a training method, an apparatus and a device.

[0004] According to a first aspect of the present disclosure, provided is a method for identifying an object sequence in an image, which includes: obtaining a target image, wherein the target image comprises a target object sequence formed by stacking a plurality of objects in a first direction; and determining a category of each of the plurality of objects in the target object sequence based on a neural network; wherein the neural network is obtained by training based on a target loss, which is determined based on a first difference between features of a sample image and features of a first auxiliary image, and a second difference between the features of the sample image and features of a second auxiliary image, and wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the features of the

sample image, the features of the first auxiliary image and the features of the second auxiliary image are extracted by the neural network.

[0005] In some embodiments, the target loss is positively correlated with the first difference within a range of first preset loss value, and the target loss is negatively correlated with the second difference within the range of first preset loss value.

[0006] In some embodiments, the target loss includes: a first loss determined based on an offset between a predicted result and an actual result for a category of the sample object sequence in the sample image; and a second loss determined based on the first difference and the second difference.

[0007] In some embodiments, the second loss is determined based on the following: determining an edit distance between the sample object sequence in the sample image and the second object sequence in the second auxiliary image, wherein the edit distance is used to represent a number of transformations required for the sample object sequence to transform to the second object sequence according to a designated transformation way, and the designated transformation way comprises deletion of objects, addition of objects and replacement of objects; and determining the second loss based on the first difference, the second difference and the edit distance.

[0008] In some embodiments, determining the second loss based on the first difference, the second difference and the edit distance includes: determining a weight parameter corresponding to the second difference based on the edit distance; and determining the second loss based on a difference between the first difference and a value obtained by weighting the second difference using the weight parameter.

[0009] In some embodiments, determining the weight parameter corresponding to the second difference based on the edit distance includes: determining the weight parameter corresponding to the second difference to be 1 in response to determining that the edit distance is greater than a preset distance; and determining the weight parameter corresponding to the second difference to be a value positively correlated with the edit distance and not greater than 1 in response to determining that the edit distance is less than or equal to the preset distance.

[0010] In some embodiments, the second loss is positively correlated with the edit distance within a range of second preset loss value.

[0011] In some embodiments, the second loss is greater than or equal to 0.

[0012] In some embodiments, determining the first loss based on the offset between the predicted result and the actual result for the category of the sample object sequence in the sample image includes: determining a first offset between the predicted result and the actual result for the category of the sample object sequence in the sample image; determining a second offset between a predicted result and an actual result for a category of the first object sequence in the first auxiliary image; determining a third offset between a predicted result and an actual result for a category of the second object sequence in the second auxiliary image; and determining the first loss based on the first offset, the second offset and the third offset.

[0013] In some embodiments, the first difference is represented by a distance between a vector corresponding to the features of the sample image and a vector corresponding to the features of the first auxiliary image; and the second difference is represented by a distance between the vector corresponding to the features of the sample image and a vector corresponding to the features of the second auxiliary image.

[0014] In some embodiments, the plurality of objects in the target object sequence include sheet objects, and the first direction includes a thickness direction of the sheet objects.

[0015] In some embodiments, a surface of each of the plurality of objects in the target object sequence in the first direction is provided with identification information, which includes one or more of a color, a pattern or a texture.

[0016] In some embodiments, the method further includes: determining, upon identifying the category of each of the plurality of objects in the target object sequence in the target image, a total face value corresponding to the target object sequence based on a face value corresponding to each category.

[0017] According to a second aspect of the present disclosure, provided is a method for training a neural network, which includes: obtaining a sample image, a first auxiliary image and a second auxiliary image, wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the sample object sequence, the first object sequence and the second object sequence are each formed by stacking a plurality of objects in a first direction; extracting, with a neural network, features of the sample image, features of the first auxiliary image, and features of the second auxiliary image; establishing a target loss based on a first difference between the features of the sample image and the features of the first auxiliary image and a

second difference between the features of the sample image and the features of the second auxiliary image; and training the neural network with the target loss as an optimization objective.

[0018] According to a third aspect of the present disclosure, provided is an apparatus for identifying an object sequence in an image, which include: an obtaining module configured to obtain a target image, wherein the target image comprises a target object sequence formed by stacking a plurality of objects in a first direction; and a prediction module configured to determine a category of each of the plurality of objects in the target object sequence based on a neural network; wherein the neural network is obtained by training based on a target loss, which is determined based on a first difference between features of a sample image and features of a first auxiliary image, and a second difference between the features of the sample image and features of a second auxiliary image, and wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the features of the sample image, the features of the first auxiliary image and the features of the second auxiliary image are extracted by the neural network.

[0019] According to a fourth aspect of the present disclosure, provided is an electronic device, including a processor and a memory for storing computer instructions executable by the processor, wherein the processor is configured to execute the computer instructions to implement the method according to the above-mentioned first aspect.

[0020] According to a fifth aspect of the present disclosure, provided is a computer-readable storage medium storing a computer program, when being executed by a processor, the computer program causes the processor to implement the method according to the above-mentioned first aspect.

[0021] In embodiments of the present disclosure, when training a neural network for identifying the object sequence in the image, a sample image, a first auxiliary image and a second auxiliary image may be input simultaneously. A first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, and a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image. Features of the sample image, features of the first auxiliary image and features of the second auxiliary image are extracted by the neural network. A target loss may be established based on a difference between the features of the sample image and the

features of the first auxiliary image and a difference between the features of the sample image and the features of the second auxiliary image, such that the distribution of the image features extracted by the trained neural network is more reasonable. In other words, the image features extracted by the trained neural network are sparse and can be distributed in a larger range, so that the difference between the features of different images may be more significant. In this way, similar object sequences in the images may also be identified accurately and the accuracy of identification results of the neural network may be improved.

[0022] It will be understood that the above general description and the later detailed description are exemplary and explanatory rather than limiting the present disclosure.

BRIEF DESCRIPTION OF DRAWINGS

[0023] The accompanying drawings herein are incorporated into and form part of the description, which illustrate embodiments consistent with the present disclosure and are used in conjunction with the description to illustrate the technical solutions of the present disclosure.

[0024] FIG. 1 is a schematic diagram of an object sequence in accordance with an embodiment of the present disclosure.

[0025] FIG. 2 is a schematic diagram of a process for training a neural network in accordance with an embodiment of the present disclosure.

[0026] FIG. 3 is a schematic flowchart of a method for identifying an object sequence in an image in accordance with an embodiment of the present disclosure.

[0027] FIG. 4 is a schematic flowchart of a method for training a neural network in accordance with an embodiment of the present disclosure.

[0028] FIG. 5 is a schematic diagram of a process for training a neural network in accordance with an embodiment of the present disclosure.

[0029] FIG. 6 is a schematic structural diagram of an apparatus for identifying an object sequence in an image in accordance with an embodiment of the present disclosure.

[0030] FIG. 7 is a schematic structural diagram of an electronic device in accordance with an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0031] Examples will be described in detail herein, with the illustrations thereof represented in the drawings. When the following descriptions involve the drawings, like

numerals in different drawings refer to like or similar elements unless otherwise indicated. The embodiments described in the following examples do not represent all embodiments consistent with the present disclosure. Rather, they are merely examples of devices and methods consistent with some aspects of the present disclosure as detailed in the appended claims.

[0032] The terms used herein are for the purpose of describing particular embodiments only and is not intended to be limiting of the present disclosure. As used herein, the singular forms “a”, “an”, “said” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term “and/or” includes any and all combinations of one or more of the associated listed items. In addition, the term “at least one” herein means any one of the plurality or any combination of at least two of the plurality.

[0033] It will be understood that while terms such as “first”, “second”, “third”, etc. may be used to describe various information, such information should not be limited to these terms. These terms are used only to distinguish the same type of information from each other. For example, without departing from the scope of the present disclosure, a first information may also be referred to as a second information, and similarly, a second information may also be referred to as a first information. Depending on the context, as used herein, the wording “if” may be interpreted as “while ...” or “when ...” or “in response to determining”.

[0034] In order to enable those skilled in the art to better understand the technical solutions in the embodiments of the present disclosure, and to enable the above-mentioned objects, features and advantages of the embodiments of the present disclosure to be more obvious and understandable, the technical solutions in the embodiments of the present disclosure are described in further detail below in conjunction with the accompanying drawings.

[0035] In some scenarios, it is necessary to identify an object sequence formed by stacking a plurality of objects in an image, in order to determine a category and attribute of each of the plurality of objects in the object sequence. Illustrated by the example (case) of a game scenario, many games are now becoming more and more intelligent. During playing the game, by capturing an image of the game area, an operation of a user as well as a game state may be identified from the image, and the game result may be output automatically. For example, tokens are usually used in the game process. In an example, as shown in FIG. 1, users can use tokens to place a betting area during the game process, and usually users will

place stacks and stacks of tokens into the betting area. In order to automatically identify the face values of the tokens placed by users, the face value of each token in a stack of tokens may be identified by capturing an image of the tokens, so as to automatically determine the total face value of the tokens placed by users. However, for object sequences formed by stacking such as tokens, it is difficult to accurately identify the category of each object in the object sequence because the stacked objects are often similar to each other and there are many ways to combine different objects in the stacking process.

[0036] An object sequence in an image may usually be identified by a trained neural network. For example, as shown in FIG. 2, which is a schematic diagram of a process for training the neural network through the related art, a large number of sample images including object sequences may be obtained, and the categories of the object sequences in these sample images may be pre-labelled, and then the sample images are input to the neural network, and network parameters of the neural network are continuously optimized with an offset between a predicted category for the object sequence in the sample image and an actual labelled category of the object sequence, to obtain the trained neural network, and then the trained neural network is used to identify an object sequence in an image. However, since the stacked objects are often similar to each other and there are many ways to combine different objects in the stacking process, the neural network trained in this way has poor identification accuracy, for example, for some similar object sequences, it is prone to false identification.

[0037] Based on this, an embodiment of the present disclosure provides a method for identifying an object sequence in an image. When training a neural network for identifying the object sequence in the image, a sample image, a first auxiliary image and a second auxiliary image may be input simultaneously. A first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, and a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image. Features of the sample image, features of the first auxiliary image and features of the second auxiliary image are extracted by the neural network. A target loss may be established based on a difference between the features of the sample image and the features of the first auxiliary image and a difference between the features of the sample image and the features of the second auxiliary image, such that the distribution of the image features extracted by the trained neural network is more reasonable. In other words, the image features extracted by the trained neural network are sparse and can be distributed in

a larger range, so that the difference between the features of different images may be more significant. In this way, similar object sequences in the images may also be identified accurately and the accuracy of identification results of the neural network may be improved.

[0038] The method for identifying an object sequence in an image provided by embodiments of the present disclosure may be performed by various electronic devices in which a trained neural network is deployed, for example, the electronic device may be a server, a server cluster, or a mobile terminal, etc., without limitation by embodiments of the present disclosure.

[0039] The object sequence in embodiments of the present disclosure may be a sequence formed by stacking a plurality of objects in a first direction, and the plurality of objects may be objects of the same category or different categories. These objects may be tokens, game coins, cards, various types of game props, etc. used in games, competitions, and entertainment scenarios, without limitation by embodiments of the present disclosure. For example, as shown in FIG. 1, the objects may be tokens or game coins in a game scenario, and the object sequence may be formed by stacking tokens or game coins in a thickness direction.

[0040] Specifically, the method for identifying an object sequence in an image in an embodiment of the present disclosure is shown in FIG. 3 and may include the following steps.

[0041] At step S302, a target image is obtained, wherein the target image includes a target object sequence formed by stacking a plurality of objects in a first direction.

[0042] At step S304, a category of each of the plurality of objects in the target object sequence is determined based on a neural network; wherein the neural network is obtained by training based on a target loss, which is determined based on a first difference between features of a sample image and features of a first auxiliary image, and a second difference between the features of the sample image and features of a second auxiliary image, and wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the features of the sample image, the features of the first auxiliary image and the features of the second auxiliary image are extracted by the neural network.

[0043] At step S302, the target image may be obtained, wherein the target image includes at least one object sequence formed by stacking a plurality of objects in the first direction,

wherein the target image may include one or more object sequences, the following will be illustrated as an example of including one target object sequence in the target image. In some scenarios, the target image may be an image of the object sequence captured directly by the image acquisition device. In some scenarios, the target image may also be an image obtained by cropping the image captured by the image capture device, for example, the image captured by the image capture device may be obtained first, and then the object sequence is detected by the target detection algorithm, and then the region including the target object sequence is cropped to obtain the target image.

[0044] At step S304, after obtaining the target image, the target image may be inputted into a neural network which has been pre-trained, and a category of each of the object in the target object sequence in the target image is predicted by the neural network. Where for objects of different category in the object sequence, there will be differences in their appearance, such as a colour, a texture, a pattern or other preset identifiers, and thus the neural network may perform feature extraction on the target image, and a category of each of the objects may be determined based on the extracted features. Illustrated by the example of the stacked tokens as the target object sequence, the tokens may have different face values, and the tokens with different face values have different textures, colours, and patterns, thus the neural network performs feature extraction on the image and determines the face value corresponding to each of the tokens in the object sequence based on the extracted features.

[0045] In order to obtain more accurate identification results for the object sequence, when training the neural network for identifying object sequences, a sample image, a first auxiliary image, and a second auxiliary image may be input, wherein the sample image, the first auxiliary image, and the second auxiliary image each include an object sequence, wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, and a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image. The number of first auxiliary images may be one frame or multiple frames and the number of second auxiliary images may be one frame or multiple frames, for example, in some scenarios, one sample image frame, one first auxiliary image frame, and one second auxiliary image frame may be simultaneously input into the neural network to train the neural network, and in some scenarios, one sample image frame, multiple first auxiliary image frames, and multiple second auxiliary image frame may also be simultaneously input into the neural network to train the neural network, which may be set according to the actual situation, without

limitation by embodiments of the present disclosure. Identical object sequences mean that the number, category and arrangement of objects in the object sequences are the same. Differences in any of the above three attributes are different object sequences. The first auxiliary image and the sample image may be captured for the same object sequence at different viewing angles, different lighting or different scenarios.

[0046] In order to make the distribution of features extracted by the neural network more reasonable, that is, the features may be distributed in a larger space, so as to make a difference between features of different object sequences more obvious, so as to facilitate the subsequent classification of objects in the object sequence based on the features, to this end, when establishing target loss for training the neural network, the target loss may be determined based on a difference between the features of the sample image and the features of the first auxiliary image and a difference between the features of the sample image and the features of the second auxiliary image, and then the neural network is trained with the target loss as the optimization target to optimize parameters of the neural network.

[0047] Since the features of the sample image, the features of the first auxiliary image, and the features of the second auxiliary image are extracted by the neural network, the target loss may be constrained based on the difference between the features of the sample image and the features of the first auxiliary image and the difference between the features of the sample image and the features of the second auxiliary image, such that the lower similar between the object sequences in the images is, the greater the difference between the features extracted by the neural network is, so that the distribution of the features extracted by the neural network is more reasonable and the difference between the features of different object sequences is as significant as possible, the accuracy of identification results of the neural network is improved, and the neural network is allowed to identify the object sequences with more combinatorial ways.

[0048] The first direction in embodiments of the present disclosure may be a length direction, a height direction, or a width direction of the objects, without limitation by embodiments of the present disclosure. For example, in some embodiments, a plurality of objects in an object sequence may be sheet objects, and that first direction includes a thickness direction of the sheet objects. For example, the objects may be tokens, and the first direction may be a thickness direction of the tokens.

[0049] In some embodiments, a surface in the first direction of each of the objects in the target object sequence is provided with identification information, which comprises one or

more of a color, a pattern and a texture, preset identifiers etc., such that the neural network may determine a category of each of the objects in the object sequence based on the identification information. Illustrated by the example of a token as an object, the object sequence may be formed by tokens stacked in the thickness direction (e.g., FIG. 1), and thus, after an image including the stacked tokens is acquired, each of the tokens may be identified based on the colour, texture, or pattern of a surface in the thickness direction of the token.

[0050] In some embodiments, in order to improve the accuracy of identification results of the neural network for the object sequence, the training process of the neural network may include the following steps, as shown in FIG. 4.

[0051] At step S402, a sample image, a first auxiliary image and a second auxiliary image are obtained, wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the sample object sequence, the first object sequence and the second object sequence are each formed by a plurality of objects stacked in a first direction.

[0052] At step S404, features of the sample image, features of the first auxiliary image, and features of the second auxiliary image are extracted by means of a neural network.

[0053] At step S406, a target loss is established based on a first difference between the features of the sample image and the features of the first auxiliary image and a second difference between the features of the sample image and the features of the second auxiliary image.

[0054] At step S408, the neural network is trained with the target loss as an optimization objective.

[0055] First, a large number of images that include a plurality of objects stacked along a first direction to form an object sequence may be obtained as training samples, each of which includes the sample image, the first auxiliary image, and the second auxiliary image. The first object sequence in the first auxiliary image is the same as the sample object sequence in the sample image, with some differences in the acquisition angle and/or acquisition environment between the sample image and the first auxiliary image. The second object sequence in the second auxiliary image is different from the sample object sequence in the sample image. During training the neural network, the above three images are simultaneously input to the neural network. Wherein, the sample image, the first auxiliary image and the second auxiliary image all carry label information, which is used to indicate

a category of each of the objects in the object sequences in the images, and an actual result for the category of the object sequence in the image may be obtained based on the label information, the actual result is an actual category sequence corresponding to actual category of various objects in the object sequence.

[0056] After obtaining the sample image, the first auxiliary image and the second auxiliary image, the features of the sample image, the features of the first auxiliary image and the features of the second auxiliary image may be extracted by the neural network. Wherein, a commonly used classification network may be select as the neural network, for example, resnet50. Usually the features extracted by the neural network are a two-dimensional matrix (H, C), where H represents a dimension of the features in the first direction (i.e., in the stacking direction), H is generally determined based on a size of a single object in the stacking direction, and C represents a number of channels of the features.

[0057] After determining the features of the sample image, the features of the first auxiliary image, and the features of the second auxiliary image, a target loss may be established based on a first difference between the features of the sample image and the features of the first auxiliary image and a second difference between the features of the sample image and the features of the second auxiliary image, and network parameters of the neural network may be continuously adjusted with the target loss as an optimization objective, to train the neural network.

[0058] In some embodiments, during establishing the target loss, a predicted result for the category of the sample object sequence in the sample image may be determined based on the features of the sample image, for example, for the identification of such sequences, a sequence identification structure (e.g., Connectionist Temporal Classification, CTC) may usually be used to determine the final predicted result, and then the target loss may be established by combining an offset between the actual result and the predicted result of the category of the sample object sequence in the sample image, the first difference and the second difference together.

[0059] In some embodiments, the features of the sample image, the features of the first auxiliary image, and the features of the second auxiliary image each may be represented by a vector. Thus, the first difference may be represented by a distance between the vector corresponding to the features of the sample image and the vector corresponding to the features of the first auxiliary image, and the second difference may be represented by a distance between the vector corresponding to the features of the sample image and the vector

corresponding to the features of the second auxiliary image. The smaller distance between the two vectors is, that is, the smaller the difference is, and the greater the similarity is, where the distance may be a Euclidean distance, a Manhattan distance, and so on, without limitation by embodiments of the present disclosure. For example, usually after extracting the features of the image by the neural network, the two-dimensional feature matrix (H, C) may be obtained, wherein, H represents the dimension of the features in the first direction (i.e., in the stacking direction), H is generally determined based on the size of a single object in the stacking direction, and C represents the number of channels of the features. In order to facilitate determining the difference of two features, a two-dimensional feature matrix may be transformed into a one-dimensional vector, such as HC, and then the distance between the two vectors is calculated for representing the difference between the features. It will be understood that, in some embodiments, the difference between the two feature matrices may also be calculated directly, which may be set according to the actual needs.

[0060] In some embodiments, when determining the target loss based on the offset between the predicted result and the actual result of the category of the sample object sequence in the sample image, the first difference between the features of the sample image and the features of the first auxiliary image, and the difference between the features of the sample image and the features of the second auxiliary image, the first loss may be determined based on the offset between the deviation result and the actual result of the category of the sample object sequence in the sample image, and the second loss is determined based on the first difference and the second difference, and the target loss is determined based on the first loss and the second loss. The first loss is used to represent the offset between the current predicted result from the neural network and the actual result, and the second loss is used to represent the difference between the features extracted by the neural network of images that include different object sequences. The second loss may be regarded as a penalty term to constrain the final target loss, so that the neural network may make the offset between the predicted result and the actual result as small as possible during the training process, and also make the features of the images with more similar object sequences extracted by the neural network more similar and the features of the images with more difference object sequences less similar, and so that the difference between the features of the images including different object sequences is more significant for easy differentiation.

[0061] In some embodiments, within a preset range of first loss value, the target loss is positively correlated with the first difference, and the target loss is negatively correlated with the second difference. When the neural network performs feature extraction, for two frames of images including the same object sequence, the more similar the extracted features are, i.e., the smaller the difference is, the smaller the loss is; and for two frames of images including different object sequences, the less similar the extracted features are, i.e., the greater the difference is, the smaller the loss is. Thus, when establishing the target loss, the target loss is allowed to be positively correlated with the first difference and negatively correlated with the second difference. It will be understood that, since the difference between the features of different two object sequences cannot be enlarged indefinitely, and therefore, the target loss may be allowed to meet the above rule within the preset range of first loss value, wherein, the range of first loss value may be flexibly set in combination with the need.

[0062] In some embodiments, the second loss is not less than 0. In the training process of the neural network, it is necessary to pay attention to the offset between the predicted result and the actual result to ensure that the predicted result is as close as possible to the actual result. Therefore, the second loss may be greater than or equal to 0 to avoid adjusting the target loss too small when the offset between the predicted result and the actual result is large, resulting in less accurate predicted result.

[0063] In some embodiments, the second loss may be determined based on the first difference and the second difference only. For example, in some embodiments, the second loss may be determined based on Formula (1):

$$\text{Loss2}=\text{Max}(\rho1-u\rho2,0) \quad \text{Formula (1)}.$$

[0064] Wherein, Max represents a maximum value, Loss2 represents the second loss, $\rho1$ represents the first difference, which may be represented by the distance between the vector corresponding to the features of the sample image and the vector corresponding to the features of the first auxiliary image; $\rho2$ represents the second difference, which may be represented by the distance between the vector corresponding to the features of the sample image and the vector corresponding to the features of the second auxiliary image, and u represents an adjustment parameter.

[0065] In some embodiments, when determining the second loss based on the first difference and the second difference, an edit distance between the sample object sequence in the sample image and the second object sequence in the second auxiliary image is

determined, such that the second loss is determined based on the first difference and the second difference in combination with the edit distance. Wherein the edit distance is used to represent a number of transformations required for the sample object sequence to transform to the second object sequence according to a designated transformation way, and the designated transformation way comprises deletion of objects, addition of objects and replacement of objects. Illustrated by the example of tokens as objects, and the tokens include four categories of a , b , c , and d , for example, the sample object sequence in the sample image is $aabc$, and the second object sequence in the second auxiliary image is $aabd$, and the number of transformations required for the sample object sequence to transform to the second object sequence is 1, i.e., c is replaced with d , and thus, the edit distance is 1. For Another example, the sample object sequence in the sample image is $abbc$, the second object sequence in the second auxiliary image is abc , and the number of transformations required for the sample object sequence to transform to the second object sequence is 2, that is, d and b in the sample object sequence are deleted, and thus, the edit distance is 2. Therefore, the edit distance may represent the similarity of two object sequences, and the more similar the two object sequences are, the smaller the edit distance is. Therefore, when determining the second loss based on the first difference and the second difference, the second loss may be further determined in combination with the edit distance. The second loss is determined based on the first difference and the second difference in combination with the edit distance, such that the strength of optimizing the feature distribution is further improved.

[0066] For example, in some embodiments, when determining the second loss based on the first difference and the second difference, the effect of the second difference on the second loss may be adjusted based on the edit distance, for example, a weight parameter corresponding to the second difference may be determined based on the edit distance, and then the second loss may be determined based on a difference value between the first difference and the second difference weighted using the weight parameter, such that the second loss may be automatically adjusted based on the edit distance.

[0067] In some embodiments, when determining the weight parameter corresponding to the second difference based on the edit distance, if the edit distance is greater than a preset distance, the weight parameter corresponding to the second difference is determined to be 1; If the edit distance is less than or equal to the preset distance, the weight parameter corresponding to the second difference is determined to be a value that is positively

correlated with the edit distance and not greater than 1, i.e., the greater the edit distance is, the greater the weight parameter is. Wherein, the preset distance may be adjusted based on actual needs. In some embodiments, within the preset range of second loss value, the second loss may be positively correlated with the edit distance, i.e., the greater the edit distance is, the greater the second loss is, such that the less similar the object sequences in the images are, the greater the difference between the features of the images is, wherein the range of second loss value may be set flexibly in combination with the needs. For example, in some embodiments, the second loss may be determined based on Formula (2):

$$\text{Loss2} = \text{Max}(\rho_1 - \lambda \rho_2, 0) \quad \text{Formula (2), wherein}$$

$$\lambda = \begin{cases} 1 & d > k \\ \frac{d}{k} & d \leq k \end{cases} .$$

[0068] Wherein Max represents a maximum value, Loss2 represents the second loss, ρ_1 represents the first difference, which may be represented by the distance between the vector corresponding to the features of the sample image and the vector corresponding to the features of the first auxiliary image; ρ_2 represents the second difference, which may be represented by the distance between the vector corresponding to the features of the sample image and the vector corresponding to the features of the second auxiliary image, d represents an edit distance between the sample object sequence in the sample image and the second object sequence in the second auxiliary image, and k is a fixed parameter.

[0069] It will be understood that, the above are only examples, and other ways of determining the second loss which are meet that the greater the edit distance is, the greater the second loss is, may be used.

[0070] In some embodiments, when determining the first loss, the first loss may be determined based on an offset between a predicted result and an actual result of the category of the sample object sequence in the sample image, i.e., only the predicted result of the sample image is considered when determining the first loss. In some embodiments, when determining the first loss, the first loss may also be determined in combination with an offset between an predicted result from the neural network and an actual results of the respective object sequences in the sample image, the first auxiliary image and the second auxiliary image. For example, a first offset between the predicted result and the actual result for the category of the sample object sequence in the sample image is determined; a second offset between a predicted result and an actual result for a category of the first object sequence in

the first auxiliary image is determined; a third offset between a predicted result and an actual result for a category of the second object sequence in the second auxiliary image is determined; and the first loss is determined based on the first offset, the second offset and the third offset. For example, the first loss may be obtained based on an average result of the first offset, the second offset, and the third offset, or a weight may be set for the above three offsets, for example, the weight of the first offset may be greater than the second offset and the third offset, and then the second loss is obtained based on the weighted average result of the first offset, the second offset, and the third offset.

[0071] In some embodiments, the objects in the object sequence may be tokens, game coins, and other objects having a face value, and each category of object corresponds to a face value, and after the category of each object of the target object sequence in the target image is determined, and the total face value corresponding to the target object sequence may also be determined based on the face values corresponding to various categories of objects.

[0072] In addition, embodiments of the present disclosure provide a training method of a neural network, which includes the following steps, as shown in FIG. 4.

[0073] At step S402, a sample image, a first auxiliary image and a second auxiliary image are obtained, wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the sample object sequence, the first object sequence and the second object sequence are each formed by a plurality of objects stacked in a first direction.

[0074] At step S404, features of the sample image, features of the first auxiliary image, and features of the second auxiliary image are extracted by means of a neural network.

[0075] At step S406, a target loss is established based on a first difference between the features of the sample image and the features of the first auxiliary image and a second difference between the features of the sample image and the features of the second auxiliary image.

[0076] At step S408, the neural network is trained with the target loss as an optimization objective.

[0077] Wherein, the specific training process of the neural network may be referred to the description in the above-described embodiments and will not be repeated herein.

[0078] In order to further explain the method for identifying the object sequence in the image in the embodiments of the present disclosure, the method will be further described with reference to the following embodiment.

[0079] In some entertainment or game scenarios, game coins are mostly used to place bets, and usually the different face values of game coins have different colours, textures and patterns. During the game, an image acquisition device may be used to capture the image of the game coin and automatically identify the category of the game coin and count the face value of the game coin based on the image. When training the neural network for identifying the game coin sequence in the image, in the related art, generally a loss function is established by an offset between a predicted result from the neural network and an actual result of the game coin sequence in the sample image, and then the neural network is trained, and the predicted result of the neural network trained in this way is not accurate enough and the accuracy is low. However, the present embodiment provides a method for identifying a game coin sequence in an image, which specifically includes a neural network training stage and a neural network prediction stage.

[0080] Neural network training stage:

[0081] The training process of the neural network may be referred to FIG. 5. Triple data may be maintained in advance, and each group of triple data includes a sample image A, a sample image B, and a sample image C. A game coin sequence b in the sample image B is the same as a game coin sequence a in the sample image A, and a game coin sequence c in the sample image C is different from the game coin sequence a in the sample image A.

[0082] The neural network may use a common classification network, such as resnet50, which may be used to extract features of an image and then input the features into a classifier for determining a category of each game coin in the game coin sequence in the image. Wherein, the game coin sequence may be identified by using a common sequence identification structure, such as CTC. During the training process, the sample image A, the sample image B, and the sample image C may be input to the neural network simultaneously, and the neural network extracts image features to obtain features fa, fb, and fc, respectively. After the features of the images are extracted by the neural network, a two-dimensional feature matrix may be obtained (H, C), wherein, H represents a dimensionality of the features in the stacking direction, H is generally determined based on a thickness of a single game coin, and C represents a number of channels of the features. In order to expediently determine a difference between two features, a two-dimensional feature matrix may be

transformed into a one-dimensional vector, such as HC, and then a distance between the two vectors is calculated for representing the difference between the features. A loss function may be established based on the features fa , fb , and fc , as in Formula (3):

$$\begin{aligned} loss &= \max(d(fa, fb) - \lambda(a, c) * d(fa, fc) + margin, 0) \\ \lambda(a, c) &= \begin{cases} 1, & edit(a, c) > k \\ edit(a, c) / k, & edit(a, c) \leq k \end{cases} \end{aligned} \quad \text{Formula (3).}$$

[0083] Wherein $edit(a, c)$ represents an edit distance between sequences a and c , $d(fa, fb)$ represents a distance between the vector of feature fa and the vector of feature fb , and $d(fa, fc)$ represents a distance between the vector of feature fa and the vector of feature fc , and both K and $margin$ are adjustable parameters.

[0084] The loss makes that the distance between the game coin sequences a and b of the same category and combination is closer, and the distance between the game coin sequences a and c of different categories and combinations is further apart. In addition, an automatically adjusted weight parameter $\lambda(a, c)$ is set for the game coin sequences a and c , so that the loss penalty is positively correlated with the difference between the game coin sequences a and c , that is, the greater the difference between the game coins in the game coin sequences a and c is, the greater $\lambda(a, c)$ is, the greater the loss is, and the features fa and fc will be further apart; and when most game coins in the game coin sequences a and c are the same and a few of them are different, the loss will be reduced accordingly, and the smaller the loss is, the closer the features fa and fc is, such that the features extracted by the neural network conform to an intuitive and reasonable feature distribution.

[0085] After determining the loss, the loss' may be further determined based on an offset between a respective predicted result and an actual result of each of the sample image A, the sample image B, and the sample image C. An final loss function may be determined based on Formula (4):

$$\text{total loss} = \text{loss} + \text{loss}' \quad \text{Formula (4).}$$

[0086] Then, the network parameters of the neural network may be optimized based on the total loss to obtain the trained neural network.

[0087] Neural network prediction stage:

[0088] After the neural network is trained, the target image may be input into the neural network, so that the category of each game coin of the game coin sequence in the target image may be determined and the total face value of the game coin sequence may be counted.

[0089] The present embodiment adds the loss determined based on the features f_a , f_b , and f_c when establishing the loss function, such that the game coin feature distribution is sparse, and the optimization strength of the loss guided by the editing distance is improved, the feature distribution is more reasonable, and the trained neural network has more accurate.

[0090] Corresponding to the method described above, embodiments of the present disclosure also provide an apparatus for identifying an object sequence in an image, as shown in FIG. 6, the apparatus 60 includes: an obtaining module 61 configured to obtain a target image, wherein the target image comprises a target object sequence formed by stacking a plurality of objects in a first direction; and a prediction module 62 configured to determine a category of each of the plurality of objects in the target object sequence based on a neural network; wherein the neural network is obtained by training based on a target loss, which is determined based on a first difference between features of a sample image and features of a first auxiliary image, and a second difference between the features of the sample image and features of a second auxiliary image, and wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the features of the sample image, the features of the first auxiliary image and the features of the second auxiliary image are extracted by the neural network.

[0091] In some embodiments, the target loss is positively correlated with the first difference within a range of first preset loss value, and the target loss is negatively correlated with the second difference within the range of first preset loss value.

[0092] In some embodiments, the target loss includes: a first loss determined based on an offset between a predicted result and an actual result for a category of the sample object sequence in the sample image; and a second loss determined based on the first difference and the second difference.

[0093] In some embodiments, the second loss is determined based on the following: determining an edit distance between the sample object sequence in the sample image and the second object sequence in the second auxiliary image, wherein the edit distance is used to represent a number of transformations required for the sample object sequence to transform to the second object sequence according to a designated transformation way, and the designated transformation way comprises deletion of objects, addition of objects and

replacement of objects; and determining the second loss based on the first difference, the second difference and the edit distance.

[0094] In some embodiments, determining the second loss based on the first difference, the second difference and the edit distance includes: determining a weight parameter corresponding to the second difference based on the edit distance; and determining the second loss based on a difference between the first difference and a value obtained by weighting the second difference using the weight parameter.

[0095] In some embodiments, determining the weight parameter corresponding to the second difference based on the edit distance includes: determining the weight parameter corresponding to the second difference to be 1 in response to determining that the edit distance is greater than a preset distance; and determining the weight parameter corresponding to the second difference to be a value positively correlated with the edit distance and not greater than 1 in response to determining that the edit distance is less than or equal to the preset distance.

[0096] In some embodiments, the second loss is positively correlated with the edit distance within a range of second preset loss value.

[0097] In some embodiments, the second loss is greater than or equal to 0.

[0098] In some embodiments, determining the first loss based on the offset between the predicted result and the actual result for the category of the sample object sequence in the sample image includes: determining a first offset between the predicted result and the actual result for the category of the sample object sequence in the sample image; determining a second offset between a predicted result and an actual result for a category of the first object sequence in the first auxiliary image; determining a third offset between a predicted result and an actual result for a category of the second object sequence in the second auxiliary image; and determining the first loss based on the first offset, the second offset and the third offset.

[0099] In some embodiments, the first difference is represented by a distance between a vector corresponding to the features of the sample image and a vector corresponding to the features of the first auxiliary image; and the second difference is represented by a distance between the vector corresponding to the features of the sample image and a vector corresponding to the features of the second auxiliary image.

[0100] In some embodiments, the plurality of objects in the target object sequence include sheet objects, and the first direction includes a thickness direction of the sheet objects.

[0101] In some embodiments, a surface of each of the plurality of objects in the target object sequence in the first direction is provided with identification information, which includes one or more of a color, a pattern or a texture.

[0102] In some embodiments, the method further includes: determining, upon identifying the category of each of the plurality of objects in the target object sequence in the target image, a total face value corresponding to the target object sequence based on a face value corresponding to each category.

[0103] In addition, as shown in FIG. 7, an embodiment of the present disclosure provides an electronic device 70, including a processor 71 and a memory 72 for storing computer instructions executable by the processor 71, wherein the processor is configured to execute the computer instructions to implement the method according to any of the above-mentioned embodiments.

[0104] An embodiment of the present disclosure provides a computer-readable storage medium storing a computer program, when being executed by a processor, the computer program causes the processor to implement the method according to any of the above-mentioned embodiments.

[0105] A computer readable medium includes permanent and non-permanent and transitory, removable and non-removable media and may be implemented by any method or technique for storing information. The information may be computer readable instructions, data structures, modules of programs, or other data. Examples of computer readable medium include, but are not limited to, Phase Change Memory (PRAM), Static Random Access Memory (SRAM), Dynamic Random Access Memory (DRAM), other types of Random Access Memory (RAM), Read-Only Memory (ROM), Electrically Erasable Programmable Read-Only Memory (EEPROM), flash memory or other memory technologies, read-only compact disc only Read Memory (CD-ROM), Digital Versatile Disc (DVD) or other optical storage, magnetic cartridge tape, magnetic tape disk storage or other magnetic storage device, or any other non-transport medium that may be used to store information that may be accessed by a computing device. As defined herein, computer readable media does not include transient computer readable media (transitory media), such as modulated data signals and carriers.

[0106] As can be seen from the above description of embodiments, it is clear to those skilled in the art that embodiments of the present specification may be implemented with software plus the necessary common hardware platform. Based on such an understanding,

the technical solution of the embodiments of the present specification that essentially, or in part, contributes to the prior art may be embodied in the form of a software product that may be stored in a storage medium, such as ROM/RAM, disk, CD-ROM, etc., comprising a number of instructions to enable a computer device (which may be a personal computer, a server, or a network device, etc.) to execute the embodiments or some embodiments of the specification.

[0107] Systems, devices, modules, or units illustrated in the above embodiments may be implemented specifically by a computer chip or entity, or by a product having some functionality. An exemplary implementation device is a computer, and the specific form of the computer may be a personal computer, a laptop computer, a cellular phone, a camera phone, a smart phone, a personal digital assistant, a media player, a navigation device, an email sending and receiving device, a gaming console, a tablet computer, a wearable device, or a combination of any of these devices.

[0108] Various embodiments in the specification are described in a progressive manner, the same or similar parts between the various embodiments may refer to each other, and each embodiment focusing on what is different from the other embodiments. In particular, since the apparatus embodiment is basically similar to the method embodiment, a brief description of the device embodiment is provided., the relevant parts may refer to the method embodiment. The apparatus embodiment described above are merely schematic, where the modules described as separate components may or may not be physically separated, and the functions of the modules may be implemented in the same or more software and/or hardware when implementing the embodiments of the specification. It is also possible to select some or all of these modules to achieve the purpose of the embodiments according to actual needs, and those skilled in the art can understand and implement embodiments of the present disclosure without creative work.

WHAT IS CLAIMED IS:

1. A method for identifying an object sequence in an image, comprising:

obtaining a target image, wherein the target image comprises a target object sequence formed by stacking a plurality of objects in a first direction; and

determining a category of each of the plurality of objects in the target object sequence based on a neural network;

wherein the neural network is obtained by training based on a target loss, which is determined based on a first difference between features of a sample image and features of a first auxiliary image, and a second difference between the features of the sample image and features of a second auxiliary image, and wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the features of the sample image, the features of the first auxiliary image and the features of the second auxiliary image are extracted by the neural network.

2. The method according to claim 1, wherein the target loss is positively correlated with the first difference within a range of first preset loss value, and the target loss is negatively correlated with the second difference within the range of first preset loss value.

3. The method according to claim 1 or 2, wherein the target loss comprises:

a first loss determined based on an offset between a predicted result and an actual result for a category of the sample object sequence in the sample image; and

a second loss determined based on the first difference and the second difference.

4. The method according to claim 3, wherein the second loss is determined based on the following:

determining an edit distance between the sample object sequence in the sample image and the second object sequence in the second auxiliary image, wherein the edit distance is used to represent a number of transformations required for the sample object sequence to transform to the second object sequence according to a designated transformation way, and the designated transformation way comprises deletion of objects, addition of objects and replacement of objects; and

determining the second loss based on the first difference, the second difference and the edit distance.

5. The method according to claim 4, wherein determining the second loss based on the first difference, the second difference and the edit distance comprises:

determining a weight parameter corresponding to the second difference based on the edit distance; and

determining the second loss based on a difference between the first difference and a value obtained by weighting the second difference using the weight parameter.

6. The method according to claim 5, wherein determining the weight parameter corresponding to the second difference based on the edit distance comprises:

determining the weight parameter corresponding to the second difference to be 1 in response to determining that the edit distance is greater than a preset distance; and

determining the weight parameter corresponding to the second difference to be a value positively correlated with the edit distance and not greater than 1 in response to determining that the edit distance is less than or equal to the preset distance.

7. The method according to any one of claims 4 to 6, wherein the second loss is positively correlated with the edit distance within a range of second preset loss value.

8. The method according to any one of claims 3 to 7, wherein the second loss is greater than or equal to 0.

9. The method according to any one of claims 3 to 8, wherein determining the first loss based on the offset between the predicted result and the actual result for the category of the sample object sequence in the sample image comprises:

determining a first offset between the predicted result and the actual result for the category of the sample object sequence in the sample image;

determining a second offset between a predicted result and an actual result for a category of the first object sequence in the first auxiliary image;

determining a third offset between a predicted result and an actual result for a category of the second object sequence in the second auxiliary image; and

determining the first loss based on the first offset, the second offset and the third offset.

10. The method according to any one of claims 1 to 9, wherein

the first difference is represented by a distance between a vector corresponding to the features of the sample image and a vector corresponding to the features of the first auxiliary image; and

the second difference is represented by a distance between the vector corresponding to the features of the sample image and a vector corresponding to the features of the second auxiliary image.

11. The method according to any one of claims 1 to 10, wherein the plurality of objects in the target object sequence comprise sheet objects, and the first direction comprises a thickness direction of the sheet objects.

12. The method according to any one of claims 1 to 11, wherein a surface of each of the plurality of objects in the target object sequence in the first direction is provided with identification information, which comprises one or more of a color, a pattern or a texture.

13. The method according to any one of claims 1 to 12, further comprising:

determining, upon identifying the category of each of the plurality of objects in the target object sequence in the target image, a total face value corresponding to the target object sequence based on a face value corresponding to each category.

14. A method for training a neural network, comprising:

obtaining a sample image, a first auxiliary image and a second auxiliary image, wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the sample object sequence, the first object sequence and the second object sequence are each formed by stacking a plurality of objects in a first direction;

extracting, with a neural network, features of the sample image, features of the first auxiliary image, and features of the second auxiliary image;

establishing a target loss based on a first difference between the features of the sample image and the features of the first auxiliary image and a second difference between the features of the sample image and the features of the second auxiliary image; and
training the neural network with the target loss as an optimization objective.

15. An apparatus for identifying an object sequence in an image, comprising:

an obtaining module configured to obtain a target image, wherein the target image comprises a target object sequence formed by stacking a plurality of objects in a first direction; and

a prediction module configured to determine a category of each of the plurality of objects in the target object sequence based on a neural network;

wherein the neural network is obtained by training based on a target loss, which is determined based on a first difference between features of a sample image and features of a first auxiliary image, and a second difference between the features of the sample image and features of a second auxiliary image, and wherein a first object sequence in the first auxiliary image is the same as a sample object sequence in the sample image, a second object sequence in the second auxiliary image is different from the sample object sequence in the sample image, and the features of the sample image, the features of the first auxiliary image and the features of the second auxiliary image are extracted by the neural network.

16. An electronic device, comprising a processor and a memory for storing computer instructions executable by the processor, wherein the processor is configured to execute the computer instructions to implement the method according to any one of claims 1 to 14.

17. A computer-readable storage medium storing a computer program, when being executed by a processor, the computer program causes the processor to implement the method according to any one of claims 1 to 14.