

TRAINING METHODS AND APPARATUSES FOR OBJECT DETECTION SYSTEM

TECHNICAL FIELD

[0001] One or more embodiments of this specification relate to the field of machine
5 learning technologies, and in particular, to training methods and apparatuses for an
object detection system.

BACKGROUND

[0002] The object detection technology aims to identify one or more objects in an
image and locate different objects (give bounding boxes). Object detection is used in
10 many scenarios such as self-driving and security systems.

[0003] Currently, mainstream object detection algorithms are mainly based on a
deep learning model. However, existing related algorithms can hardly satisfy increasing
needs in actual applications. Therefore, an object detection solution is needed, so that
accuracy of a detection result can be ensured while a calculation amount is reduced, to
15 better satisfy needs in actual applications.

SUMMARY

[0004] One or more embodiments of this specification describe training methods
for an object detection system. A new object detection algorithm architecture is
designed by introducing both convolutional layers and attention layers into a backbone
20 network, to relieve dependence of a deep learning architecture on pre-training, and
effectively reduce a calculation amount needed to train the object detection system.

[0005] According to a first aspect, a training method for an object detection system
is provided. The object detection system includes a backbone network and a head
network, the backbone network includes several convolutional layers and several self-
25 attention layers, and the method includes the following: a training image is input to the
object detection system, where convolution processing is performed on the training

image by using the several convolutional layers, to obtain a convolution representation; self-attention processing is performed based on the convolution representation by using the several attention layers, to obtain a feature map; and the feature map is processed by using the head network, to obtain a detection result of a target object in the training
5 image; a gradient norm of each neural network layer is determined based on object annotation data and the detection result corresponding to the training image; and for each neural network layer, network parameters of the neural network layer are updated based on an average of the gradient norms and the gradient norm of the neural network layer.

10 **[0006]** In one embodiment, the detection result includes a classification result and a detection bounding box of the target object, and the object annotation data include an object classification result and an object annotation bounding box.

[0007] In one embodiment, the convolution representation includes C two-dimensional matrices, and the performing self-attention processing based on the
15 convolution representation by using the several attention layers, to obtain a feature map includes the following: self-attention processing is performed, by using the several attention layers, on C vectors obtained by performing flattening processing based on the C two-dimensional matrices, to obtain Z vectors; and truncation and stack processing is respectively performed on the Z vectors to obtain Z two-dimensional
20 matrices as the feature map.

[0008] In one embodiment, the head network includes a region proposal network (RPN) and a classification and regression layer, and the processing the feature map by using the head network, to obtain a detection result of a target object in the training image includes the following: a plurality of proposed regions that include the target
25 object are determined by using the RPN based on the feature map; and a target object category and a bounding box that correspond to each proposed region are determined by using the classification and regression layer based on a region feature of the proposed region, and the target object category and the bounding box are used as the detection result.

30 **[0009]** In one embodiment, the determining a gradient norm of each neural network layer based on object annotation data and the detection result corresponding to the training image includes the following: a gradient of each neural network layer is calculated based on the object annotation data and the detection result by using a back propagation method; and a norm of the gradient of each neural network layer is

calculated as a corresponding gradient norm.

[0010] In one embodiment, the object detection system includes a plurality of neural network layers, and the updating, for each neural network layer, network parameters of the neural network layer based on an average of the gradient norms and the gradient norms of the neural network layer includes the following: an average of a plurality of gradient norms corresponding to the plurality of neural network layers is calculated; and for each neural network layer, the network parameters of the neural network layer are updated based on a ratio of the gradient norm of the neural network layer to the average.

[0011] In one specific embodiment, the calculating an average of a plurality of gradient norms corresponding to the plurality of neural network layers includes the following: a geometric mean of the plurality of gradient norms is calculated.

[0012] In one specific embodiment, the updating, for each neural network layer, the network parameters of the neural network layer based on a ratio of the gradient norm of the neural network layer to the average includes the following: for each neural network layer, the ratio of the gradient norm of the neural network layer to the average is calculated; an exponentiation result obtained by using the ratio as the base and a predetermined value as the exponent is determined; and the network parameters of the neural network layer are updated to a product of the network parameters and the exponentiation result.

[0013] According to a second aspect, a training apparatus for an object detection system is provided. The object detection system includes a backbone network and a head network, the backbone network includes several convolutional layers and several self-attention layers, and the apparatus includes the following: an image processing unit, configured to process a training image by using the object detection system, where the image processing unit includes the following: a convolution subunit, configured to perform convolution processing on the training image by using the several convolutional layers, to obtain a convolution representation; an attention subunit, configured to perform self-attention processing based on the convolution representation by using the several attention layers, to obtain a feature map; and a processing subunit, configured to process the feature map by using the head network, to obtain a detection result of a target object in the training image; a gradient norm calculation unit, configured to determine a gradient norm of each neural network layer based on object annotation data and the detection result corresponding to the training image; and a

network parameter update unit, configured to update, for each neural network layer, network parameters of the neural network layer based on an average of the gradient norms and the gradient norm of the neural network layer.

[0014] According to a third aspect, a computer-readable storage medium is provided, where the computer-readable storage medium stores a computer program, and when the computer program is executed on a computer, the computer is enabled to perform the method according to the first aspect.

[0015] According to a fourth aspect, a computing device is provided, including a memory and a processor, where the memory stores executable code, and when executing the executable code, the processor implements the method according to the first aspect.

[0016] According to the methods and the apparatuses provided in the embodiments of this specification, the backbone network in the object detection system is configured as a hybrid architecture including convolutional layers and self-attention layers. In addition, a gradient fine-tuning technique is proposed to adjust the training gradients of each neural network layer in the object detection system, so that good precision can also be achieved by directly performing single-stage training without performing pre-training on the object detection system.

BRIEF DESCRIPTION OF DRAWINGS

[0017] To describe the technical solutions in the embodiments of this application more clearly, the following briefly describes the accompanying drawings needed for describing the embodiments. Clearly, the accompanying drawings in the following descriptions show merely some embodiments of this application, and a person of ordinary skill in the art can still derive other drawings from these accompanying drawings without creative efforts.

[0018] FIG. 1 is a schematic diagram illustrating a system architecture of an object detection system, according to an embodiment;

[0019] FIG. 2 is a schematic flowchart illustrating a training method for an object detection system, according to an embodiment;

[0020] FIG. 3 is a schematic structural diagram illustrating a self-attention block in a Transformer mechanism;

[0021] FIG. 4 is a schematic diagram illustrating a process of processing an image

by using an object detection system, according to an embodiment; and

[0022] FIG. 5 is a schematic structural diagram illustrating a training apparatus for an object detection system, according to an embodiment.

DESCRIPTION OF EMBODIMENTS

5 [0023] The following describes the solutions provided in this specification with reference to the accompanying drawings.

[0024] As described above, current mainstream object detection algorithms are mainly based on a deep learning architecture. However, because the deep learning model has a large number of parameters, a current object detector generally needs two
10 steps of training to achieve good precision. The two steps of training include pre-training and fine-tuning. Pre-training generally takes a long time to perform training in a very large data set (for example, an ImageNet data set), and consumes a very large number of computing resources. Fine-tuning is to briefly train a pre-trained model in a target data set (such as a COCO data set and actual service data), so that the model fits
15 the data.

[0025] Popular deep learning architectures include a convolutional neural network (CNN) and a Transformer. Because pre-training excessively consumes time and computing resources, in the era when the CNN network was the mainstream detector framework, many researchers have explored how to achieve a good detection effect
20 while discarding pre-training. Unfortunately, their success cannot be replicated in the transformer architecture, that is, it is currently not possible to train a Transformer-based detector to have good precision without pre-training.

[0026] Further, the inventor finds that the convolutional layer in the CNN network has an inductive bias that can be understood as a prior knowledge. Generally, a stronger
25 prior knowledge indicates weaker dependence on pre-training. The inductive bias of the CNN network includes locality, that is, there is a relationship between pixel blocks with spatial positions close to each other and there is no relationship between pixel blocks with spatial positions far from each other, and includes spatial invariance, for example, a tiger is a tiger either on the left or the right of an image. In addition, the self-attention
30 layer in the Transformer allows for a global attention mechanism that consumes a large number of compute and strongly depends on pre-training. However, in the pre-training phase, a self-attention layer near the input end actually determines the inductive bias,

and behaves like a convolution operation.

[0027] Based on this, the inventor proposes to replace the first several self-attention layers close to the input end in the Transformer-based deep learning architecture with convolutional layers, thereby directly reducing dependence of the Transformer-based detector on pre-training.

[0028] FIG. 1 is a schematic diagram illustrating a system architecture of an object detection system, according to an embodiment. As shown in FIG. 1, the object detection system includes a backbone network and a head network. The backbone network is used to perform encoding representation on an image, and includes several convolutional layers and several self-attention layers that are respectively shown as m convolutional layers and n self-attention layers in FIG. 1. The head network is used to determine an object detection box and a classification category based on the encoding representation. It should be understood that "several" in this specification means one or more, and values of m and n can be set and adjusted based on actual needs.

[0029] However, network structures of the convolutional layer and the attention layer differ greatly, and a good effect can hardly be achieved by directly performing training based on a conventional method. In practice, the inventor finds that a gradient of the attention layer is ten times higher than a gradient of the convolutional layer, and therefore proposes a gradient fine-tuning technique, so that the above-mentioned object detection system can obtain good training performance.

[0030] FIG. 2 is a schematic flowchart illustrating a training method for an object detection system, according to an embodiment. The method can be performed by any platform, server, or device cluster that has a calculation and processing capability. As shown in FIG. 2, the method includes the following steps.

[0031] Step S210: Input a training image to the object detection system. Specifically, in substep S211, convolution processing is performed on the training image by using several convolutional layers, to obtain a convolution representation. In substep S212, self-attention processing is performed based on the convolution representation by using several attention layers, to obtain a feature map. In substep S213, the feature map is processed by using the head network, to obtain a detection result of a target object in the training image. Step S220: Determine a gradient norm of each neural network layer based on object annotation data and the detection result corresponding to the training image. Step S230: Update, for each neural network layer, network parameters of the neural network layer based on an average of the gradient norms and the gradient norm

of the neural network layer.

[0032] The above-mentioned steps are described in detail as follows:

[0033] First, in step S210, the training image is input to the object detection system, and the detection result of the target object in the training image is output. Specifically,
5 step S210 includes the following substeps:

[0034] Step S211: Perform convolution processing on the training image by using the several convolutional layers, to obtain the convolution representation.

[0035] It is worthwhile to note that convolution processing (or a convolution

operation) is a commonly used operation when an image is analyzed. Through

10 convolution processing, abstract features can be extracted from a pixel matrix of an

original image. Based on a design of a convolution kernel, these abstract features can

reflect, for example, more global features such as a line shape and color distribution of

a region in the original image. Further, convolution processing means using several

convolution kernels in a single convolutional layer to perform convolution calculation

15 on an image representation (usually a three-dimensional tensor) that is input to the layer.

Specifically, when convolution calculation is performed, each of the several

convolution kernels is slid over a feature matrix corresponding to a height dimension

and a width dimension in the image representation. For each stride, a product of each

element in the convolution kernel and a value of a matrix element covered by the

20 element is multiplied, and the products are summed. As such, a new image

representation can be obtained.

[0036] Each of the several convolutional layers, that is, one or more convolutional

layers, performs convolution processing on an image representation output by a

previous convolutional layer of the convolutional layer, so that an image representation

25 output by the last convolutional layer is used as the above-mentioned convolution

representation. It can be understood that an input of the first convolutional layer is an

original training image.

[0037] In one embodiment, a rectified linear unit (ReLU) activation layer is further

disposed between some of the several convolutional layers or after a certain

30 convolutional layer, to perform non-linear mapping on an output result of the

convolutional layer. A result of non-linear mapping can be input to a next convolutional

layer for further convolution processing, or can be output as the above-mentioned

convolution representation. In other embodiments, a pooling layer is further disposed

between some convolutional layers, to perform a pooling operation on an output result

of the convolutional layer. The result of the pooling operation can be input to a next convolutional layer to continue to perform a convolution operation. In still other embodiments, a residual block is further disposed after a certain convolutional layer. The residual block performs addition processing on an input and an output of the certain convolutional layer, and uses a result of the addition processing as an input of a next convolutional layer or the ReLU activation layer.

[0038] In the above-mentioned descriptions, one or more convolutional layers can be used, and the ReLU activation layer and/or the pooling layer can be selectively added based on needs, to process the above-mentioned training image and obtain a corresponding convolution representation.

[0039] Step S212: Perform self-attention processing based on the convolution representation by using the several attention layers, to obtain the feature map.

[0040] It is worthwhile to note that an output of the convolutional layer and an input of the attention layer generally have different data formats. Therefore, the convolution representation needs to be reshaped, and then the reshaped convolution representation is used as an input of the attention layer. Specifically, the convolution representation is generally a three-dimensional tensor, and can be denoted as (W, H, C) , where W and H respectively correspond to a width dimension and a height dimension of an image, and C is a number of channels. In this case, the convolution representation can also be considered as C two-dimensional matrices. However, the input of the attention layer needs to be a vector sequence. Therefore, it is proposed to perform flattening processing on the W dimension and the H dimension. That is, for each of the C two-dimensional matrices, row vectors in the matrix are sequentially spliced to obtain a corresponding one-dimensional vector, so that C $(W*H)$ -dimensional vectors can be obtained, to form a vector sequence. Therefore, the vector sequence can be used as an input of the first attention layer in the several attention layers. In addition, both formats of an input and an output of the attention layer are vector sequences, or the input and the output can be considered as matrices forming vector sequences.

[0041] The above-mentioned self-attention processing is a processing method where a self-attention mechanism is introduced. The self-attention mechanism is one of attention mechanisms. When processing information, a human selectively pays attention to a part of all information, and ignores other visible information. This mechanism is generally referred to as the attention mechanism, and the self-attention mechanism means that external information is not introduced when existing

information is processed. For example, when each word in a sentence is encoded by using the self-attention mechanism, only information about all words in the sentence is referenced, and text content other than the sentence is not introduced.

[0042] In this step, a self-attention processing method in the Transformer
5 mechanism can be used for reference. Specifically, for any i^{th} attention layer in the several attention layers, an input matrix of the i^{th} attention layer can be denoted as $Z^{(i)}$. Therefore, for the i^{th} attention layer, the matrix $Z^{(i)}$ is first respectively projected to a query space, a key space, and a value space, to obtain a query matrix Q, a key matrix K, and a value matrix V. Then, an attention weight is determined by using the query
10 matrix Q and the key matrix K, and the value matrix V is transformed by using the determined attention weight, so that a matrix $Z^{(i+1)}$ obtained through transformation is used as an output of the current attention layer.

[0043] In one embodiment, a residual block and a feedforward layer can be further designed to form a self-attention block together with the self-attention layer, to process
15 the above-mentioned convolution representation. FIG. 3 is a schematic structural diagram illustrating a self-attention block in a Transformer mechanism. As shown in FIG. 3, the self-attention block includes an attention layer, a residual block, a feedforward layer, and another residual block that are sequentially connected. The self-attention layer first processes a matrix $Z^{(i)}$ input to the layer, to obtain a matrix output
20 by the self-attention layer. The residual block R1 first performs addition processing on the output matrix and the above-mentioned matrix $Z^{(i)}$, and then performs normalization processing. The feedforward layer performs linear transformation and non-linear transformation on an output of the residual block R1, and an output of the layer continues to be processed by the residual block R2, to obtain an output matrix $Z^{(i+1)}$ of
25 the current self-attention block. Further, if the current self-attention block is followed by another self-attention block, the output matrix can be used as an input of the next attention block. Otherwise, the above-mentioned feature map can be determined based on the output matrix.

[0044] In the above-mentioned descriptions, a matrix (or a vector sequence) output
30 by each self-attention layer or each self-attention block can be obtained, to determine the above-mentioned feature map. In one embodiment, the feature map can be determined based on an output of the last self-attention layer in the several self-attention layers or based on an output of the last self-attention block in the several self-attention blocks. In other embodiments, the feature map can be determined based on an average

matrix of all matrices output by all self-attention layers or all self-attention blocks.

[0045] Further, a reverse operation corresponding to the above-mentioned flattening processing is performed on the output vector sequence, to obtain the feature map. Specifically, for each vector in the vector sequence, the vector is truncated to a predetermined number of sub-vectors that have the same length as each other, and then the sub-vectors are stacked to obtain a corresponding two-dimensional matrix. Therefore, S two-dimensional matrices corresponding to a plurality of (that can be denoted as S) vectors included in the vector sequence can be obtained, to form the feature map.

[0046] In the above-mentioned descriptions, self-attention processing can be performed on the convolution representation, to obtain the feature map of the training image.

[0047] Step S213: Process the feature map by using the head network, to obtain the detection result of the target object in the training image.

[0048] It is worthwhile to note that for the head network, a head network in an anchor-based object detection algorithm such as a faster region-based convolutional neural network (Faster-RCNN) or a feature pyramid network (FPN) can be used, or a head network in an anchor-free object detection algorithm can be used. The head network in the classic Faster-RCNN algorithm is used as an example below to describe implementation of the step.

[0049] FIG. 4 is a schematic diagram illustrating a process for processing an image by using an object detection system, according to an embodiment. The head network includes a region proposal network (RPN) and a classification and regression layer shown in the figure.

[0050] Specifically, a plurality of proposed regions (RP) that include the target object are first determined by using the RPN based on the feature map. The proposed region is a region where an object may appear in an image. In some cases, the proposed region is also referred to as a region of interest. The proposed region is determined to provide a basis for subsequent object classification and determining of regression of a bounding box. As shown in the example in FIG. 4, in an example, the RPN recommends region bounding boxes of three proposed regions in the feature map, and the region bounding boxes are respectively represented as regions A, B, and C.

[0051] Then, the feature map and generation results of the plurality of proposed regions based on the feature map are input to the classification and regression layer. For

each proposed region, the classification and regression layer determines an object category and a bounding box in the proposed region based on a region feature of the proposed region.

[0052] Based on one implementation, the classification and regression layer is a fully-connected layer, and object category classification and bounding box regression are performed based on a region feature of each region input to a previous layer. More specifically, the classification and regression layer can include a plurality of classifiers, the classifiers are trained to identify objects of different categories in a proposed region. In an animal detection scenario, the classifiers are trained to identify animals of different categories such as a tiger, a lion, a starfish, and a swallow.

[0053] The classification and regression layer further includes a regressor used to perform regression on a bounding box corresponding to an identified object, and determine that a minimum rectangular region surrounding the object is a bounding box.

[0054] Therefore, the detection result of the training image can be obtained, including a classification result and a detection bounding box of the target object.

[0055] After the training image is processed by using the object detection system to obtain the corresponding detection result, step S220 is performed to determine the gradient norm of each neural network layer based on the object annotation data and the detection result corresponding to the training image. It should be understood that the object detection system includes a plurality of neural network layers. The neural network layer is generally a network layer that includes weight parameters to be determined, for example, the self-attention layer and the convolutional layer in the backbone network.

[0056] As mentioned above, network structures of the convolutional layer and the attention layer differ greatly, and a good effect can hardly be achieved by performing training based on a conventional method. Therefore, a gradient fine-tuning technique is proposed. Specifically, there is a large difference between a gradient of the attention layer and a gradient of the convolutional layer, and actual experience shows that minor adjustment of parameters of all network layers result in a trained model with a better effect than large adjustment of parameters of certain network layers. Therefore, the inventor proposes that, after the gradient of each network layer in the object detection system is calculated, parameter adjustment is not directly performed by using an original gradient. Instead, an average of the gradient norms of all the neural network layers in the object detection system is calculated, and it is determined, based on the

average, whether the gradient of each network layer is large or small, and a magnitude is determined. Then, the network parameters of each layer are adjusted based on the obtained deviation magnitude, so that the parameters are close to the obtained average.

[0057] In one embodiment, a gradient of each neural network layer can be calculated based on the object annotation data and the detection result corresponding to the training image by using a back propagation method. Then, a norm of the gradient of each neural network layer is calculated as a corresponding gradient norm. The object annotation data include an object classification result and an object annotation bounding box, and can be obtained through manual marking. In other embodiments, a gradient of one neural network layer can be calculated to trigger calculation of a gradient norm without waiting for all gradients of all the layers to be calculated before calculation of the gradient norm.

[0058] Gradient calculation can be implemented by using an existing technology. For calculation of the gradient norm, a first-order norm, a second-order norm, etc. can be calculated. Based on an example, the following equation (1) can be used to calculate a gradient norm $C_{i,j}$ of parameters in a j^{th} neuron in any i^{th} network layer, and then a gradient norm C_i corresponding to the i^{th} network layer is calculated based on the equation (2):

$$C_{i,j} = \mathbb{E}[(z_{i-1} * y_i(j))^2] \quad (1)$$

$$C_i = \mathbb{E}_j[C_{i,j}] \quad (2)$$

[0059] In the equation (1), z_{i-1} represents an output of an activation function in an $(i-1)^{\text{th}}$ neural network layer, $y_i(j)$ represents a back propagation error of the j^{th} neuron in the i^{th} network layer, and a calculation result of $z_{i-1} * y_i(j)$ is the gradient of the parameters in the j^{th} neuron in the i^{th} network layer.

[0060] Therefore, the gradient norm C_i of each neural network layer can be determined.

[0061] Then, in step S230, for each neural network layer, the network parameters of the neural network layer are updated based on the gradient norm of the neural network layer and an average of a plurality of gradient norms corresponding to the plurality of neural network layers.

[0062] In one embodiment, an arithmetic mean of the plurality of gradient norms can be calculated, that is, the gradient norms are summed and then divided by a total number. In other embodiments, a geometric mean of the plurality of gradient norms can

be calculated, that is, the plurality of gradient norms are multiplied and then the n^{th} root is taken, where n is equal to the total number. This operation can be performed according to the following equation (3):

$$\bar{C} = (\prod_i C_i)^{\frac{1}{N}} \quad (3)$$

5 **[0063]** In one embodiment, for each neural network layer, a ratio of the gradient norm C_i of the neural network layer to the average $\bar{C}$ is calculated to update the network parameters of the neural network layer based on the ratio. In one specific embodiment, an exponentiation result obtained by using the ratio as the base and a predetermined value α (for example, $\alpha = 0.25$) as the exponent can be first
10 determined, that is, $r_i = (C_i/\bar{C})^\alpha$. Further, the network parameters W_i of the neural network layer are updated to a product of the network parameters and the exponentiation result, and can be denoted as $W_i \leftarrow r_i W_i$. In other specific embodiments, the network parameters of the neural network layer can be directly updated to a product of the network parameters and the ratio. As such, the network parameters of the object
15 detection system can be effectively updated.

[0064] In conclusion, according to the training methods for an object detection system disclosed in the embodiments of this specification, the backbone network in the object detection system is configured as a hybrid architecture including convolutional layers and self-attention layers. In addition, a gradient fine-tuning technique is proposed
20 to adjust the training gradients of each neural network layer in the object detection system, so that good precision can also be achieved by directly performing single-stage training without performing pre-training on the object detection system.

[0065] Corresponding to the above-mentioned training method, the embodiments of this specification further disclose a training apparatus. FIG. 5 is a schematic structural diagram illustrating a training apparatus for an object detection system,
25 according to an embodiment. The object detection system includes a backbone network and a head network, and the backbone network includes several convolutional layers and several self-attention layers. As shown in FIG. 5, the apparatus 500 includes the following: an image processing unit 510, configured to process a training image by
30 using the object detection system, where the image processing unit 510 includes the following: a convolution subunit 511, configured to perform convolution processing on the training image by using the several convolutional layers, to obtain a convolution representation; an attention subunit 512, configured to perform self-attention

processing based on the convolution representation by using the several attention layers, to obtain a feature map; and a processing subunit 513, configured to process the feature map by using the head network, to obtain a detection result of a target object in the training image; a gradient norm calculation unit 520, configured to determine a gradient norm of each neural network layer based on object annotation data and the detection result corresponding to the training image; and a network parameter update unit 530, configured to update, for each neural network layer, network parameters of the neural network layer based on an average of the gradient norms and the gradient norm of the neural network layer.

10 **[0066]** In one embodiment, the detection result includes a classification result and a detection bounding box of the target object, and the object annotation data include an object classification result and an object annotation bounding box.

15 **[0067]** In one embodiment, the convolution representation includes C two-dimensional matrices, and the attention subunit 512 is specifically configured to perform, by using the several attention layers, self-attention processing on C vectors obtained by performing flattening processing based on the C two-dimensional matrices, to obtain Z vectors; and respectively perform truncation and stack processing on the Z vectors to obtain Z two-dimensional matrices as the feature map.

20 **[0068]** In one embodiment, the head network includes an RPN and a classification and regression layer, and the processing subunit 513 is specifically configured to determine, by using the RPN based on the feature map, a plurality of proposed regions that include the target object; and determine, by using the classification and regression layer based on a region feature of each proposed region, a target object category and a bounding box that correspond to the proposed region, and use the target object category and the bounding box as the detection result.

25 **[0069]** In one embodiment, the gradient norm calculation unit 520 is specifically configured to calculate a gradient of each neural network layer based on the object annotation data and the detection result by using a back propagation method; and calculate a norm of the gradient of each neural network layer as a corresponding gradient norm.

30 **[0070]** In one embodiment, the object detection system includes a plurality of neural network layers, and the network parameter update unit 530 includes the following: an average calculation subunit 531, configured to calculate an average of a plurality of gradient norms corresponding to the plurality of neural network layers; and

a parameter update subunit 532, configured to update, for each neural network layer, the network parameters of the neural network layer based on a ratio of the gradient norm of the neural network layer to the average.

[0071] In one embodiment, the average calculation subunit 531 is specifically
5 configured to calculate a geometric mean of the plurality of gradient norms.

[0072] In one embodiment, the parameter update subunit 532 is specifically configured to calculate, for each neural network layer, the ratio of the gradient norm of the neural network layer to the average; determine an exponentiation result obtained by using the ratio as the base and a predetermined value as the exponent; and update the
10 network parameters of the neural network layer to a product of the network parameters and the exponentiation result.

[0073] In conclusion, according to the training apparatuses for an object detection system disclosed in the embodiments of this specification, the backbone network in the object detection system is configured as a hybrid architecture including convolutional
15 layers and self-attention layers. In addition, a gradient fine-tuning technique is proposed to adjust the training gradients of each neural network layer in the object detection system, so that good precision can also be achieved by directly performing single-stage training without performing pre-training on the object detection system.

[0074] In embodiments of another aspect, a computer-readable storage medium is
20 further provided. The computer-readable storage medium stores a computer program. When the computer program is executed on a computer, the computer is enabled to perform the method described with reference to FIG. 2.

[0075] In embodiments of still another aspect, a computing device is further provided, including a memory and a processor. The memory stores executable code,
25 and when executing the executable code, the processor implements the method the method described with reference to FIG. 2. A person skilled in the art should be aware that in the above-mentioned one or more examples, functions described in this application can be implemented by hardware, software, firmware, or any combination thereof. When this application is implemented by software, the functions can be stored
30 in a computer-readable medium or transmitted as one or more instructions or code in the computer-readable medium.

[0076] The objectives, technical solutions, and beneficial effects of this application are further described in detail in the above-mentioned specific implementations. It should be understood that the above-mentioned descriptions are merely specific

implementations of this application, but are not intended to limit the protection scope of this application. Any modification, equivalent replacement, or improvement made based on the technical solutions of this application shall fall within the protection scope of this application.

5

CLAIMS

What is claimed is:

1. A training method for an object detection system, wherein the object detection system comprises a backbone network and a head network, the backbone network
5 comprises several convolutional layers and several self-attention layers, and the method comprises:

inputting a training image to the object detection system, wherein
convolution processing is performed on the training image by using the several
convolutional layers, to obtain a convolution representation; self-attention processing
10 is performed based on the convolution representation by using the several attention
layers, to obtain a feature map; and the feature map is processed by using the head
network, to obtain a detection result of a target object in the training image;

determining a gradient norm of each neural network layer based on object
annotation data and the detection result corresponding to the training image; and
15 updating, for each neural network layer, network parameters of the neural network
layer based on an average of the gradient norms and the gradient norm of the neural
network layer.

2. The method according to claim 1, wherein the detection result comprises a
classification result and a detection bounding box of the target object, and the object
20 annotation data comprise an object classification result and an object annotation
bounding box.

3. The method according to claim 1, wherein the convolution representation
comprises C two-dimensional matrices, and the performing self-attention processing
based on the convolution representation by using the several attention layers, to obtain
25 a feature map comprises:

performing, by using the several attention layers, self-attention processing on C
vectors obtained by performing flattening processing based on the C two-dimensional
matrices, to obtain Z vectors; and

respectively performing truncation and stack processing on the Z vectors to obtain
30 Z two-dimensional matrices as the feature map.

4. The method according to claim 1, wherein the head network comprises a region
proposal network (RPN) and a classification and regression layer, and the processing
the feature map by using the head network, to obtain a detection result of a target object

in the training image comprises:

determining, by using the RPN based on the feature map, a plurality of proposed regions that comprise the target object; and

5 determining, by using the classification and regression layer based on a region feature of each proposed region, a target object category and a bounding box that correspond to the proposed region, and using the target object category and the bounding box as the detection result.

10 5. The method according to claim 1, wherein the determining a gradient norm of each neural network layer based on object annotation data and the detection result corresponding to the training image comprises:

calculating a gradient of each neural network layer based on the object annotation data and the detection result by using a back propagation method; and

calculating a norm of the gradient of each neural network layer as a corresponding gradient norm.

15 6. The method according to claim 1, wherein the object detection system comprises a plurality of neural network layers, and the updating, for each neural network layer, network parameters of the neural network layer based on an average of the gradient norms and the gradient norm of the neural network layer comprises:

20 calculating an average of a plurality of gradient norms corresponding to the plurality of neural network layers; and

updating, for each neural network layer, the network parameters of the neural network layer based on a ratio of the gradient norm of the neural network layer to the average.

25 7. The method according to claim 6, wherein the calculating an average of a plurality of gradient norms corresponding to the plurality of neural network layers comprises:

calculating a geometric mean of the plurality of gradient norms.

30 8. The method according to claim 6, wherein the updating, for each neural network layer, the network parameters of the neural network layer based on a ratio of the gradient norm of the neural network layer to the average comprises:

for each neural network layer, calculating the ratio of the gradient norm of the neural network layer to the average; determining an exponentiation result obtained by using the ratio as the base and a predetermined value as the exponent; and updating the network parameters of the neural network layer to a product of the network parameters

and the exponentiation result.

9. A training apparatus for an object detection system, wherein the object detection system comprises a backbone network and a head network, the backbone network comprises several convolutional layers and several self-attention layers, and the apparatus comprises:

an image processing unit, configured to process a training image by using the object detection system, wherein the image processing unit comprises: a convolution subunit, configured to perform convolution processing on the training image by using the several convolutional layers, to obtain a convolution representation; an attention subunit, configured to perform self-attention processing based on the convolution representation by using the several attention layers, to obtain a feature map; and a processing subunit, configured to process the feature map by using the head network, to obtain a detection result of a target object in the training image;

a gradient norm calculation unit, configured to determine a gradient norm of each neural network layer based on object annotation data and the detection result corresponding to the training image; and

a network parameter update unit, configured to update, for each neural network layer, network parameters of the neural network layer based on an average of the gradient norms and the gradient norm of the neural network layer.

10. A computing device, comprising a memory and a processor, wherein the memory stores executable code, and when executing the executable code, the processor implements the method according to any one of claims 1 to 8.