

DEEP MODEL TRAINING METHOD AND APPARATUS, ELECTRONIC DEVICE, AND STORAGE MEDIUM

CROSS-REFERENCE TO RELATED APPLICATION

[0001] The present disclosure is based upon and claims priority to Chinese Patent Application No. 201811646430.5 filed on December 29, 2018, the disclosure of which is hereby incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] The present disclosure generally relates to, but is not limited to, the field of information technology, and more particularly to a method and an apparatus for training a deep model, an electronic device and a storage medium.

BACKGROUND

[0003] A deep learning model may have certain classification or recognition capabilities after being trained through a training set. The training set generally includes training data and annotation data of the training data. However, in general, annotation data is obtained through annotating the data manually. Annotating all the training data purely by hand is a heavy workload with low efficiency, and manual errors exist in the annotation process. Besides, if high-accuracy annotation such as annotations in the image field is required, it is required to achieve pixel-level segmentation. However, it is very difficult to achieve pixel-level segmentation and to guarantee annotation accuracy by means of pure manual annotation.

[0004] Therefore, the training of the deep learning model based on pure manually-annotated training data will have low training efficiency, and the accuracy of classification or recognition of the model obtained through the training will not reach the expectation due to the low accuracy of the training data itself.

SUMMARY

- [0005] In view of the above, the embodiments of the present disclosure are intended to provide a method and an apparatus for training a deep model, an electronic device and a storage medium.
- [0006] The technical solutions of the present disclosure are implemented as follows.
- [0007] A first aspect of the embodiments of the present disclosure provides a method for training a deep learning model, including:
- [0008] obtaining (n+1)th annotation information output by a model to be trained, herein the model to be trained has undergone n rounds of training, where n is an integer greater than or equal to 1;
- [0009] generating an (n+1)th training sample based on training data and the (n+1)th annotation information; and
- [0010] performing an (n+1)th round of training on the model to be trained using the (n+1)th training sample.
- [0011] Based on the above solution, generating the (n+1)th training sample based on the training data and the (n+1)th annotation information includes:
- [0012] generating the (n+1)th training sample based on the training data, the (n+1)th annotation information and a first training sample; or
- [0013] generating the (n+1)th training sample based on the training data, the (n+1)th annotation information and an nth training sample, herein the nth training sample comprises: a first training sample composed of the training data and first annotation information, and a second training sample to an (n-1)th training sample respectively composed of annotation information obtained through previous n-1 rounds of training and training samples used in the previous n-1 rounds of training.
- [0014] Based on the above solution, the method further includes:

- [0015] determining whether n is less than N , where N is a maximum number of training rounds of the model to be trained,
- [0016] herein obtaining the $(n+1)$ th annotation information output by the model to be trained includes:
- [0017] obtaining the $(n+1)$ th annotation information output by the model to be trained, responsive to n being less than N .
- [0018] Based on the above solution, the method further includes:
- [0019] obtaining the training data and initial annotation information of the training data;
and
- [0020] generating the first annotation information based on the initial annotation information.
- [0021] Based on the above solution, obtaining the training data and the initial annotation information of the training data includes:
- [0022] obtaining a training image containing multiple segmentation objects and a bounding box of each segmentation object,
- [0023] herein generating the first annotation information based on the initial annotation information includes:
- [0024] drawing, within the bounding box, an annotation contour consistent with a shape of the segmentation object based on the bounding box.
- [0025] Based on the above solution, generating the first annotation information based on the initial annotation information further includes:
- [0026] generating a segmentation boundary of two of the segmentation objects based on bounding boxes of the segmentation objects, the two segmentation objects having an overlapping part.
- [0027] Based on the above solution, drawing, within the bounding box, the annotation contour consistent with the shape of the segmentation object based on the bounding box

includes:

- [0028] drawing, within the bounding box, an inscribed ellipse of the bounding box consistent with a shape of a cell based on the bounding box.
- [0029] A second aspect of the embodiments of the present disclosure provides an apparatus for training a deep learning model, including:
 - [0030] an annotation module, configured to obtain (n+1)th annotation information output by a model to be trained, herein the model to be trained has undergone n rounds of training, where n is an integer greater than or equal to 1;
 - [0031] a first generating module, configured to generate an (n+1)th training sample based on training data and the (n+1)th annotation information; and
 - [0032] a training module, configured to perform an (n+1)th round of training on the model to be trained using the (n+1)th training sample.
- [0033] Based on the above solution, the first generation module is configured to generate the (n+1)th training sample based on the training data, the (n+1)th annotation information and a first training sample; or generate the (n+1)th training sample based on the training data, the (n+1)th annotation information and an nth training sample, herein the nth training sample comprises: a first training sample composed of the training data and first annotation information, and a second training sample to an (n-1)th training sample respectively composed of annotation information obtained through the previous n-1 rounds of training and training samples used in the previous n-1 rounds of training.
- [0034] Based on the above solution, the apparatus includes:
 - [0035] a determining module, configured to determine whether n is less than N, where N is a maximum number of training rounds of the model to be trained,
 - [0036] where the annotation module is configured to, responsive to n being less than N, obtain (n+1)th annotation information output by the model to be trained.
- [0037] Based on the above solution, the apparatus includes:
 - [0038] an obtaining module, configured to obtain the training data and initial annotation

information of the training data; and

[0039] a second generating module, configured to generate the first annotation information based on the initial annotation information.

[0040] Based on the above solution, the obtaining module is configured to obtain a training image containing multiple segmentation objects and a bounding box of each segmentation object; and

[0041] the second generating module is configured to draw, within the bounding box, an annotation contour consistent with a shape of the segmentation object based on the bounding box.

[0042] Based on the above solution, the first generation module is configured to generate a segmentation boundary of two of the segmentation objects based on bounding boxes of the segmentation objects, the two segmentation objects having an overlapping part.

[0043] Based on the above solution, the second generation module is configured to draw, within the bounding box, an inscribed ellipse of the bounding box consistent with a shape of a cell based on the bounding box.

[0044] A third aspect of the embodiments of the present disclosure provides a computer storage medium having stored thereon computer executable instructions configured to implement the method for training a deep learning model provided by any one of the foregoing technical solutions when being executed.

[0045] A fourth aspect of the embodiments of the present disclosure provides an electronic device, including:

[0046] a memory; and

[0047] a processor, connected to the memory and configured to execute computer executable instructions stored in the memory to implement the method for training a deep learning model provided by any one of the foregoing technical solutions.

[0048] A fifth aspect of the embodiments of the present disclosure provides a computer program product, including computer executable instructions configured to implement the

method for training a deep learning model provided by any one of the foregoing technical solutions when being executed.

[0049] In the technical solutions provided by the embodiments of the present disclosure, annotation information is obtained by utilizing a deep learning model to annotate training data after a previous round of training is completed, and the annotation information is used as a training sample for a next round of training. A very small amount of initially annotated (for example, initially annotated by hand or devices) training data may be used for performing model training, and then the annotation data recognized and output by the gradually converging model to be trained itself may be used as the next round of training sample. Because model parameters of the model to be trained generated during the previous round of training will be based on the majority of data, which is annotated correctly, while a small amount of data with incorrect annotation or annotation of low accuracy has little impact on the model parameters of the model to be trained, annotation information of the model to be trained becomes more and more accurate and the training results are getting better through repeated iterations. Since the model uses its own annotation information to construct training samples, the amount of data need to be initially annotated such as annotated by hand is reduced, and the manual errors caused by initial annotation such as annotation made by hand is reduced and the efficiency is improved. Besides, a model may be trained at a fast speed and the training effect is good. The deep learning model trained by adopting this method has high accuracy in classification or recognition.

BRIEF DESCRIPTION OF THE DRAWINGS

[0050] FIG. 1 is a schematic flowchart of a first method for training a deep learning model provided by an embodiment of the present disclosure;

[0051] FIG. 2 is a schematic flowchart of a second method for training a deep learning model provided by an embodiment of the present disclosure;

[0052] FIG. 3 is a schematic flowchart of a third method for training a deep learning model provided by an embodiment of the present disclosure;

[0053] FIG. 4 is a schematic structural diagram of an apparatus for training a deep learning model provided by an embodiment of the present disclosure;

[0054] FIG. 5 is a schematic diagram of changes in a training set provided by an embodiment of the present disclosure; and

[0055] FIG. 6 is a schematic structural diagram of an electronic device provided by an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0056] The technical solutions of the present disclosure will be further described in detail below in conjunction with the drawings and specific embodiments of the specification.

[0057] As shown in FIG. 1, this embodiment provides a method for training a deep learning model, including the following operations.

[0058] In S110, (n+1)th annotation information output by a model to be trained is obtained, herein the model to be trained has undergone n rounds of training.

[0059] In S120, an (n+1)th training sample is generated based on training data and the (n+1)th annotation information.

[0060] In S130, the (n+1)th round of training is performed on the model to be trained using the (n+1)th training sample.

[0061] The method for training a deep learning model provided in this embodiment may be used in various electronic devices, for example, servers used for various big data model training.

[0062] In the first round of training, a model structure of a model to be trained is obtained. Taking a model to be trained being a neural network as an example, the network structure of the neural network needs to be figured out first. The network structure may include: the number of layers of the network, the number of nodes included in each layer, the connection relationship of the nodes among layers and the initial network parameters.

The network parameters include: a weight and/or a threshold of the node.

[0063] A first training sample is obtained, the first training sample may include: training data and first annotation data of the training data. Taking image segmentation as an example, the training data is an image, and the first annotation data may be a masked image of the segmentation object and the background in the image. In the embodiments of the present disclosure, all first annotation information and second annotation information may include, but are not limited to, annotation information of the image. The image may include medical images and the like. The medical image may be a plane (2D) medical image or a stereo (3D) medical image composed of an image sequence formed by multiple 2D images. Each of the first annotation information and the second annotation information may be annotations of an organ and/or a tissue in a medical image, or annotations of different cell structures in a cell, such as annotations of a cell nucleus. In some embodiments, the image is not limited to medical images, and can also be images of traffic road conditions when the method is applied in the field of traffic roads.

[0064] The first training sample is used to perform a first round of training on the model to be trained. After a deep learning model such as a neural network is trained, the model parameters of the deep learning model (for example, network parameters of the neural network) are changed. The model to be trained with the changed model parameters is used to process the image and output the annotation information. The annotation information is compared with the initial first annotation information, and the current loss value of the deep learning model is calculated based on the result of the comparison. If the current loss value is less than the loss threshold, this round of training can be stopped.

[0065] In S110 of the embodiment, the model to be trained that has completed n rounds of training is used to process the training data. Then, the model to be trained will obtain an output, which is the (n+1)th annotation data. A training sample is formed by associating the (n+1)th annotation data with the training data.

[0066] In some embodiments, the training data and the (n+1)th annotation information may be directly used as the (n+1)th training sample which is used as the training sample of the (n+1)th round of training of the model to be trained.

[0067] In some other embodiments, the training data, the (n+1)th annotation data, and

the first training sample may be combined to form the training sample of the (n+1)th round of training of the model to be trained.

[0068] The first training sample is a training sample used to perform the first round of training on the model to be trained; the Mth training sample is a training sample used to perform the Mth round of training on the model to be trained, where M is a positive integer.

[0069] The first training sample here may be the training data obtained initially and the first annotation information of the training data, herein the first annotation information may be manually-annotated information.

[0070] In some other embodiments, the (n+1)th training sample may be the union of the nth training sample used in the nth round of training and a training sample generated based on the training data and the (n+1)th annotation information.

[0071] In short, the above three methods for generating the (n+1)th training sample are all methods of automatically generating samples by the device. In this way, there is no need to obtain the training sample of the (n+1)th round of training by means of the manual annotation or annotation made by other devices. Therefore, the time spent for annotating the samples initially, such as manual annotation, is reduced and the speed for training the deep learning model is increased. Furthermore, the phenomena of the inaccuracy of classification or recognition results of the deep learning model after being trained due to inaccuracy caused by the manual annotation is reduced, and the accuracy of classification or recognition results of the deep learning model after being trained is increased.

[0072] Completing a round of training in this embodiment includes: the model to be trained has completed learning on each training sample in the training set at least once.

[0073] In S130, the (n+1)th training sample is used to perform the (n+1)th round of training on the model to be trained.

[0074] In this embodiment, if there are a few errors in the initial annotation, since the common characteristics of the training samples will be paid attention to during the model training process, the impact of these errors on the model training will become smaller and smaller, and the accuracy of the model will become increasingly high.

[0075] For example, taking the training data being S images as an example, the first

training sample may be S images and the result of manual annotation of these S images. If the accuracy of the annotation of one of the S images is not sufficient, but in the process of first round of training of the model to be trained, since the accuracy of the annotation of the remaining S-1 images reaches the expected threshold, the S-1 images and annotation data corresponding to the S-1 images have larger impact on the model parameters of the model to be trained. In this embodiment, the deep learning model includes, but is not limited to, a neural network, and the model parameters include, but are not limited to, a weight and/or a threshold of each network node in the neural network. The neural network may be various types of neural networks, for example, a U-net or a V-net. The neural network may include an encoding part performing feature extraction on the training data and a decoding part obtaining semantic information based on extracted features.

[0076] For example, the encoding part may perform feature extraction on a region where the segmentation object is located in the image, to obtain a masked image that distinguishes the segmentation object from the background. Based on the masked image, the decoder may obtain some semantic information, for example, omics features of the object obtained through pixel statistics, etc.

[0077] The omics features may include morphological features such as the area, volume and shape of the object, and/or gray value features formed based on the gray value.

[0078] The gray value features may include statistical characteristics of a histogram and the like.

[0079] In short, in this embodiment, when the model to be trained having undergone a first round of training recognizes S images, an image with insufficiently accurate initial annotation will have less impact on model parameters of the model to be trained than the impact made by other S-1 images. The model to be trained will use network parameters learned from other S-1 images for performing annotation. At this time, the accuracy of the annotation of the image with the insufficiently accurate initial annotation is getting closer to the accuracy of the annotation of other S-1 images, so the second annotation information corresponding to the image with the insufficiently accurate initial annotation is more accurate than the original first annotation information. In this way, the constructed second training set includes: training data composed of S images and the original first annotation information, and training data composed of S images and the second annotation

information annotated by the model to be trained itself. Therefore, in this embodiment, the negative effect of the training sample with insufficiently accurate or incorrect initial annotation will be gradually suppressed by utilizing the capability of the model to be trained to learn based on most correct or high-accurate annotation information during the training process. Thus, by adopting such a way to perform automatic iteration of the deep learning model, the manual annotation of training sample is greatly reduced, and the accuracy of the training is gradually improved through self-iteration, making the accuracy of the model to be trained after being trained achieve an expected effect.

[0080] In the above examples, images are taken as the training data. In some embodiments, the training data may also be audio clips or text information other than the images, etc. In short, the training data has many forms and is not limited to any of the above.

[0081] In some embodiments, as shown in FIG. 2, the method includes:

[0082] In S100, whether n is less than N is determined, where N is the maximum number of training rounds of the model to be trained,

[0083] the S110 further include:

[0084] responsive to n being less than N , the model to be trained obtains $(n+1)$ th annotation information output by the model to be trained.

[0085] In this embodiment, before constructing an $(n+1)$ th training set, whether or not the number of training rounds of the current model to be trained reaches the predetermined maximum number of training rounds N will be determined. If not, the $(n+1)$ th annotation information is generated to construct the $(n+1)$ th training set. Otherwise, it is determined that the model training has been completed and the training of the deep learning model is stopped.

[0086] In some embodiments, the value of N may be an empirical value or a statistical value such as 4, 5, 6, 7 or 8.

[0087] In some embodiments, the value range of N may be between 3 and 10, and the value of N may be a user input value received by a training device from a human-computer interaction interface.

- [0088] In some other embodiments, determining whether to stop the training of the model to be trained may further include the following operations.
- [0089] A test set is used to test the model to be trained. If the test result shows that the accuracy of the annotation result of the test data in the test set made by the model to be trained reaches a certain value, the training of the model to be trained is stopped, otherwise S110 is performed to enter the next round of training. At this time, the test set may be an accurately annotated data set, so the test set may be used to measure the training result of each round of training of a model to be trained to determine whether to stop the training of the model to be trained.
- [0090] In some embodiments, as shown in FIG. 3, the method includes following operations.
- [0091] In S210, the training data and initial annotation information of the training data are obtained.
- [0092] In S220, the first annotation information is generated based on the initial annotation information.
- [0093] In this embodiment, the initial annotation information may be the original annotation information of the training data. The original annotation information may be manually-annotated information or information annotated by other devices. For example, information annotated by other devices capable of performing annotation.
- [0094] In this embodiment, after the training data and the initial annotation information are obtained, first annotation information is generated based on the initial annotation information. The first annotation information here may directly include the initial annotation information and/or refined first annotation information generated according to the initial annotation information.
- [0095] For example, if the training data is an image and the image contains a cell image, the initial annotation information may be annotation information that roughly annotates the location of the cell image, while the first annotation information may be annotation information that accurately indicates the location of the cell. In short, in this embodiment, the accuracy of the annotation of the first annotation information on the segmented object

may be higher than the accuracy of the initial annotation information.

[0096] In this way, even if the initial annotation information is annotated manually, the difficulty of manual annotation is reduced, and the manual annotation is simplified.

[0097] For example, taking a cell image as an example, due to the shape of a cell being an ellipsoid, the outer contour of the cell in a two-dimensional plane image is generally elliptical. The initial annotation information may be a bounding box of the cell manually drawn by a doctor. The first annotation information may be an inscribed ellipse generated by a training device based on the manually-annotated bounding box. Compared with the bounding box, the inscribed ellipse has a reduced number of pixels in the cell image that do not belong to the cell image, thus the accuracy of the first annotation information is higher than the accuracy of the initial annotation information.

[0098] Therefore, the S210 may further include: acquiring a training image containing multiple segmentation objects and a bounding box of each segmentation object.

[0099] The S220 may include: drawing, within the bounding box, an annotation contour consistent with a shape of the segmentation object based on the bounding box.

[00100] In some embodiments, the annotation contour consistent with the shape of the segmentation object may be the aforementioned ellipse but is not limited to the ellipse. For example, it may also be a circle, a triangle or other diagonal shapes that are equal to the shape of the segmentation object.

[00101] In some embodiments, the annotation contour is inscribed in the bounding box. The bounding box may be a rectangular box.

[00102] In some embodiments, the S220 further includes:

[00103] generating a segmentation boundary of two of the segmentation objects based on bounding boxes of the segmentation objects, the two segmentation objects having an overlapping part.

[00104] In some images, two segmentation objects may overlap. In this embodiment, the first annotation information further includes a segmentation boundary between the two overlapping segmentation objects.

[00105] For example, there are two cell images A and B, and the cell image A is superimposed on the cell image B. After the cell boundary of the cell image A is drawn and the cell boundary of cell image B is drawn, the part formed by two crossed cell boundaries outlines the intersection between these two cell images. In this embodiment, according to the positional relationship between the cell image A and the cell image B, the part of the cell boundary of the cell image B located inside the cell image A may be erased, and the part of the cell boundary of the cell image A, which is located inside the cell image B, is taken as the segmentation boundary.

[00106] In short, in this embodiment, the S220 may include: drawing a segmentation boundary on the overlapping part of the two segmentation objects by utilizing the positional relationship of these two segmentation objects.

[00107] In some embodiments, drawing a segmentation boundary may be completed by modifying the boundary of one of the two segmentation objects with overlapping boundaries. In order to emphasize the boundary, the boundary may be thickened by means of pixel expansion. For example, expanding the cell boundary of the cell image A by a predetermined number of pixels, such as one or multiple pixels in a direction from the overlapping part towards the cell image B, the boundary of the cell image A of the overlapping part is thickened, so that the thickened boundary is recognized as the segmentation boundary.

[00108] In some embodiments, drawing, within the bounding box, the annotation contour consistent with the shape of the segmentation object based on the bounding box includes: drawing, within the bounding box, an inscribed ellipse of the bounding box consistent with a shape of a cell based on the bounding box.

[00109] In this embodiment, the segmentation object is a cell image, and the annotation contour includes an inscribed ellipse of the bounding box consistent with a shape of a cell.

[00110] In this embodiment, the first annotation information includes at least one of:

[00111] a cell boundary of the cell image (corresponding to the inscribed ellipse); and

[00112] a segmentation boundary between the overlapped cell image.

[00113] If, in some embodiments, the segmentation object is not a cell but other objects,

for example, the segmentation object may be faces in a group photo, the bounding box of the face may still be a rectangular box, but the annotation boundary of the face may be a boundary of an oval face, a boundary of a round face, etc. At this time, the shape is not limited to the inscribed ellipse.

[00114] Of course, the above are only examples. In short, in this embodiment, the model to be trained outputs, during its own training process, annotation information of the training data by utilizing its previous round of training result, to construct the next round of training set. Model training is completed through multiple repeated iterations without annotating a large number of training samples manually, which has a fast training rate and may improve accuracy of the training through repeated iterations.

[00115] As shown in FIG. 5, this embodiment provides an apparatus for training a deep learning model, the apparatus including:

[00116] an annotation module 110, configured to obtain (n+1)th annotation information output by a model to be trained, herein the model to be trained has undergone n rounds of training, where n is an integer greater than or equal to 1;

[00117] a first generating module 120, configured to generate an (n+1)th training sample based on training data and the (n+1)th annotation information; and

[00118] a training module 130, configured to perform an (n+1)th round of training on the model to be trained using the (n+1)th training sample.

[00119] In some embodiments, the annotation module 110, the first generating module 120 and the training module 130 may be a program module. When being executed by the processor, the program module may achieve the generation of the (n+1)th annotation information, the composition of the (n+1)th training set and the training of the model to be trained.

[00120] In some other embodiments, the annotation module 110, the first generating module 120 and the training module 130 may be a model combining software and hardware; the model combining software and hardware may be various programmable arrays, for example, a field programmable array or complex programmable array.

[00121] In other embodiments, the annotation module 110, the first generating module

120, and the training module 130 may be a pure hardware module, and the pure hardware module may be application-specific integrated circuits.

[00122] In some embodiments, the first generating module 120 is configured to generate the (n+1)th training sample based on the training data, the (n+1)th annotation information and a first training sample; or generate the (n+1)th training sample based on the training data, the (n+1)th annotation information and an nth training sample, where the nth training sample includes: a first training sample composed of the training data and first annotation information, and a second training sample to an (n-1)th training sample respectively composed of annotation information obtained through the previous n-1 rounds of training and training samples used in the previous n-1 rounds of training.

[00123] In some embodiments, the apparatus includes:

[00124] a determining module, configured to determine whether n is less than N, where N is a maximum number of training rounds of the model to be trained,

[00125] where the annotation module 110 is configured to, responsive to n being less than N, obtain (n+1)th annotation information output by the model to be trained.

[00126] In some embodiments, the apparatus includes:

[00127] an obtaining module, configured to obtain the training data and initial annotation information of the training data; and

[00128] a second generating module, configured to generate the first annotation information based on the initial annotation information.

[00129] In some embodiments, the obtaining module is configured to obtain a training image containing multiple segmentation objects and a bounding box of each segmentation object,

[00130] herein generating the first annotation information based on the initial annotation information includes:

[00131] drawing, within the bounding box, an annotation contour consistent with a shape of the segmentation object based on the bounding box.

[00132] In some embodiments, the first generating module 120 is configured to generate a segmentation boundary of two of the segmentation objects based on bounding boxes of the segmentation objects, the two segmentation objects having an overlapping part.

[00133] In some embodiments, the second generating module is configured to draw, within the bounding box, an inscribed ellipse of the bounding box consistent with a shape of a cell based on the bounding box.

[00134] A specific example is provided below in conjunction with the foregoing embodiment.

[00135] Example 1

[00136] This example provides a self-learning weak-supervised learning method of a deep learning model.

[00137] Taking the enclosing rectangular box of each object in FIG. 5 as an input, the pixel segmentation result of the object and other objects that are not annotated may be output through self-learning.

[00138] Taking cell segmentation as an example, there are enclosing rectangular annotations of some cells in the image at the beginning. It is observed that most of the cells are ellipses, so the largest inscribed ellipses are drawn in respective rectangles, segmentation lines are drawn between different ellipses, and segmentation lines are drawn on the edge of ellipses. The image and the annotation serve as the initial supervisory signal. The supervisory signal here is the training sample in the training set.

[00139] A segmentation model is trained.

[00140] Prediction is made by the segmentation model on this image, and a prediction map is obtained, the union of the prediction map and the initial annotation map serves as a new supervisory signal, and then the segmentation model is trained repeatedly.

[00141] It is found through observation that the segmentation result in the picture is getting increasingly better.

[00142] As shown in FIG. 5, the original image is annotated to obtain a masked image to construct a first training set, and the first training set is used for performing the first

round of training. After the training, the deep learning model is used for performing image recognition to obtain second annotation information, and a second training set is constructed based on the second annotation information. After the second round of training is completed using the second training set, third annotation information is output, and a third training set is obtained based on the third annotation information. After multiple rounds of training through repeated iteration, the training is stopped.

[00143] In related technologies, the probability map of the first segmentation result needs to be thoroughly studied, peak values, flat areas and the like are analyzed, and then the region growing is made. For readers, the reproduction is of a heavy workload and hard to implement. The method for training a deep learning model provided in this example does not perform any calculation on the output probability map of segmentation, but directly makes a union of the probability map of segmentation and the annotation map, and then continues to training the model. This process is simple to implement.

[00144] As shown in FIG. 6, an embodiment of the present disclosure provides an electronic device, the electronic device including:

[00145] a memory for storing information; and

[00146] a processor, connected to the memory and configured to execute computer executable instructions stored in the memory to implement the method for training a deep learning model provided by one or more of the foregoing technical solutions, for example, one or more of the methods shown in FIGS. 1 to 3.

[00147] The memory may be various types of memories, such as a random access memory, a read-only memory and a flash memory, etc. The memory may be used for information storage, for example, the memory may be used to store computer executable instructions and the like. The computer executable instructions may be various program instructions, for example, object program instructions and/or source program instructions.

[00148] The processor may be various types of processors, for example, a central processing unit, a microprocessor, a digital signal processor, a programmable array, a digital signal processor, an application-specific integrated circuit or an image processor.

[00149] The processor may be connected to the memory through a bus. The bus may be

an integrated circuit bus or the like.

[00150] In some embodiments, the terminal device may further include a communication interface. The communication interface may include a network interface, for example, a local area network interface, a transceiver antenna and the like. The communication interface is also connected to the processor and may be used for information transmission and reception.

[00151] In some embodiments, the electronic device further includes a camera, which may collect various images, for example, medical images.

[00152] In some embodiments, the terminal device further includes a human-computer interaction interface. For example, the human-computer interaction interface may include various input and output devices, such as a keyboard and a touch screen.

[00153] The embodiments of the present disclosure provide a computer storage medium having stored thereon computer executable codes configured to implement, when being executed, the method for training a deep learning model provided by one or more technical solutions, for example, one or more of the methods shown in FIGS. 1 to FIG. 3.

[00154] The storage medium includes various media capable of storing program codes, such as a mobile storage device, a read-only memory (ROM), a random access memory (RAM), a magnetic disk or an optical disk. The storage medium may be a non-transitory storage medium.

[00155] The embodiments of the present disclosure provide a computer program product, the program product includes computer executable instructions configured to implement, when being executed, the method for training a deep learning model provided by any of the foregoing implementations, for example, one or more of the methods as shown in FIGS. 1 to FIG. 3.

[00156] In the several embodiments provided in the present disclosure, it should be understood that the disclosed device and method may be implemented in other ways. The device embodiments described above are merely illustrative. For example, the division of the units is only the division of logical functions, and there may be other divisions in actual implementation, such as: multiple units or components may be combined, or be integrated

into another system, or some features can be ignored or not implemented. In addition, the coupling, or direct coupling, or communication connection between the components shown or discussed may be indirect coupling or communication connection between devices or units through some interfaces, and may be electrical, mechanical or other forms.

[00157] The units described above as separate components may or may not be physically separate, and the components displayed as units may or may not be physical units, that is, they may be located in one place or distributed on multiple network units. Some or all of the units may be selected according to actual needs to achieve the purpose of the solution of the embodiment.

[00158] In addition, the functional units in the embodiments of the present disclosure may be all integrated into one processing module, or each unit may be individually used as a unit, or two or more units can be integrated into one unit. The unit may be implemented in the form of hardware, or in the form of hardware plus software functional units.

[00159] An embodiment of the present disclosure discloses a computer program product. The program product includes computer executable instructions configured to implement, when being executed, the method for training a deep model in the foregoing embodiment.

[00160] Those of ordinary skill in the art can understand that all or part of the steps in the above method embodiments can be implemented by programs instructing relevant hardware. The foregoing programs can be stored in a computer readable storage medium. The programs execute the operations including the foregoing method embodiment when being executed. The foregoing storage medium includes various medium that can store program codes, such as removable storage devices, read-only memories (ROM), random access memories (RAM), magnetic disks or optical disks, etc.

[00161] The above are only specific implementations of the present disclosure, but the protection scope of the present disclosure is not limited thereto. Changes or substitutions that any person skilled in the art can easily think of within the technical scope disclosed in the present disclosure should be covered within the protection scope of the present disclosure. Therefore, the protection scope of the present disclosure should be subject to the protection scope of the claims.

CLAIMS

1. A method for training a deep learning model, comprising:

obtaining (n+1)th annotation information output by a model to be trained, wherein the model to be trained has undergone n rounds of training, where n is an integer greater than or equal to 1;

generating an (n+1)th training sample based on training data and the (n+1)th annotation information; and

performing an (n+1)th round of training on the model to be trained using the (n+1)th training sample.

2. The method of claim 1, wherein generating the (n+1)th training sample based on the training data and the (n+1)th annotation information comprises:

generating the (n+1)th training sample based on the training data, the (n+1)th annotation information and a first training sample; or

generating the (n+1)th training sample based on the training data, the (n+1)th annotation information and an nth training sample, wherein the nth training sample comprises: a first training sample composed of the training data and first annotation information, and a second training sample to an (n-1)th training sample respectively composed of annotation information obtained through previous n-1 rounds of training and training samples used in the previous n-1 rounds of training.

3. The method of claim 1 or 2, further comprising:

determining whether n is less than N, where N is a maximum number of training rounds

of the model to be trained,

wherein obtaining the (n+1)th annotation information output by the model to be trained comprises:

obtaining the (n+1)th annotation information output by the model to be trained, responsive to n being less than N.

4. The method of claim 2, further comprising:

obtaining the training data and initial annotation information of the training data; and

generating the first annotation information based on the initial annotation information.

5. The method of claim 4, wherein obtaining the training data and the initial annotation information of the training data comprises:

obtaining a training image containing multiple segmentation objects and a bounding box of each segmentation object,

wherein generating the first annotation information based on the initial annotation information comprises:

drawing, within the bounding box, an annotation contour consistent with a shape of the segmentation object based on the bounding box.

6. The method of claim 5, wherein generating the first annotation information based on the initial annotation information further comprises:

generating a segmentation boundary of two of the segmentation objects based on bounding boxes of the segmentation objects, the two segmentation objects having an overlapping part.

7. The method of claim 5, wherein drawing, within the bounding box, the annotation contour consistent with the shape of the segmentation object based on the bounding box comprises:

drawing, within the bounding box, an inscribed ellipse of the bounding box consistent with a shape of a cell based on the bounding box.

8. An apparatus for training a deep learning model, comprising:

an annotation module, configured to obtain (n+1)th annotation information output by a model to be trained, wherein the model to be trained has undergone n rounds of training, where n is an integer greater than or equal to 1;

a first generating module, configured to generate an (n+1)th training sample based on training data and the (n+1)th annotation information; and

a training module, configured to perform an (n+1)th round of training on the model to be trained using the (n+1)th training sample.

9. The apparatus of claim 8, wherein the first generating module is configured to: generate the (n+1)th training sample based on the training data, the (n+1)th annotation information and a first training sample; or, generate the (n+1)th training sample based on the training data, the (n+1)th annotation information and an nth training sample, wherein the nth training sample comprises: a first training sample composed of the training data and first annotation information, and a second training sample to an (n-1)th training sample respectively composed of annotation information obtained through the previous n-1 rounds of training and training samples used in the previous n-1 rounds of training.

10. The apparatus of claim 8 or 9, further comprising:

a determining module, configured to determine whether n is less than N, where N is a maximum number of training rounds of the model to be trained,

wherein the annotation module is configured to, responsive to n being less than N , obtain $(n+1)$ th annotation information output by the model to be trained.

11. The apparatus of claim 9, further comprising:

an obtaining module, configured to obtain the training data and initial annotation information of the training data; and

a second generating module, configured to generate the first annotation information based on the initial annotation information.

12. The apparatus of claim 11, wherein the obtaining module is configured to obtain a training image containing multiple segmentation objects and a bounding box of each segmentation object; and

the second generating module is configured to draw, within the bounding box, an annotation contour consistent with a shape of the segmentation object based on the bounding box.

13. The apparatus of claim 12, wherein the first generating module is configured to generate a segmentation boundary of two of the segmentation objects based on bounding boxes of the segmentation objects, the two segmentation objects having an overlapping part.

14. The apparatus of claim 12, wherein the second generating module is configured to draw, within the bounding box, an inscribed ellipse of the bounding box consistent with a shape of a cell based on the bounding box.

15. A computer storage medium having stored thereon computer executable instructions configured to implement the method of any one of claims 1 to 7 when being executed.

16. An electronic device, comprising:

a memory; and

a processor, connected to the memory and configured to execute computer executable instructions stored in the memory to implement the method of any one of claims 1 to 7.

17. A computer program product comprising computer executable instructions configured to implement the method of any one of claims 1 to 7 when being executed.