

METHOD AND APPARATUS FOR DETERMINING DEEP LEARNING NETWORK, ELECTRONIC DEVICE, AND STORAGE MEDIUM

[1] This application claims priority to and the benefit of Chinese Application Serial No. 202110150032.X, filed February 3, 2021, and entitled “METHOD AND APPARATUS FOR DETERMINING DEEP LEARNING NETWORK, ELECTRONIC DEVICE, AND STORAGE MEDIUM”, the entire disclosure of which is hereby incorporated by reference.

TECHNICAL FIELD

[2] This disclosure relates to the field of computer technology, and in particular, relates to a method and apparatus for determining deep learning network, an electronic device, and a storage medium.

BACKGROUND

[3] With the development of science and technology and the increasing maturity of human civilization, the application of deep learning network in equipment is becoming more and more common. However, the device resources provided by different devices are different. For example, the device resources provided by mobile phones and servers are obviously different. In order to use deep learning networks on different devices, researchers often need to manually design different networks for different devices. From the construction of the network structure of the deep learning network to the training of the deep learning network based on different sample data, until obtaining the trained deep learning network, which will lead researchers to waste a lot of time on network design and training and delay other work.

SUMMARY

[4] A method and apparatus for determining deep learning network, an electronic device, and a storage medium are proposed in this disclosure, which can reduce the time spent by researchers in network design and training.

[5] In some possible embodiments, a method for determining a deep learning network is provided. The method includes: determining a target event to be processed by a target device; determining an original network from a trained network set according to the target event, the original network being a network in the network set which matches the target

event; obtaining device resource information of an original device that trains the original network and device resource information of the target device; dividing a target network from the original network according to the device resource information of the target device and the device resource information of the original device, the target network being to be deployed on the target device to process the target event, the target network and the original network are the same in the number of network modules and a connection structure between network modules.

[6] In some possible embodiments, the method further include training to obtain the original network: obtaining a sample data set, the sample data set includes multiple pieces of input data and labels each corresponding to one piece of input data; constructing a preset learning network, determining multiple sampling networks from the preset learning network, and setting an initial parameter for the sampling network; training the multiple sampling networks with the sample data set and assigning the multiple sampling networks trained as multiple standby networks, and determining the original network from the multiple standby networks.

[7] In some possible embodiments, the multiple sampling networks include a first level network, at least one second level network, and a third level network, the first level network, the at least one second level network, and the third level network are same in the number of network modules and a connection structure between network modules; each network module in the first level network having an output resolution greater than a corresponding network module in the second level network, and each network module in the second level network having an output resolution greater than a corresponding network module in the third level network; each network module in the first level network having an output channel quantity greater than a corresponding network module in the second level network, and each network module in the second level network having an output channel quantity greater than a corresponding network module in the third level network; each network module in the first level network having a convolution module quantity greater than a corresponding network module in the second level network, and each network module in the second level network having a convolution module quantity greater than a corresponding network module in the third level network.

[8] In some possible embodiments, training the multiple sampling networks with the sample data set and assigning the multiple sampling networks trained as the multiple standby networks includes: assigning each of the multiple sampling networks as a present sampling network; processing the multiple pieces of input data based on the present sampling network

to determine an output data corresponding to each of the multiple pieces of input data; determining a loss value based on the output data and a label corresponding to the input data; and determining the present sampling network as a standby network when the loss value is less than or equal to a preset threshold.

[9] In some possible embodiments, the method further includes: before determining the present sampling network as the standby network when the loss value is less than or equal to the preset threshold, back propagating based on the loss value when the loss value is greater than the preset threshold, performing parameter update on the present sampling network to obtain an updated sampling network, and determining the updated sampling network as the present sampling network; repeating the following: processing the multiple pieces of input data based on the present sampling network to determine the output data corresponding to each of the multiple pieces of input data, and determining the loss value based on the output data and the label corresponding to the input data.

[10] In some possible embodiments, training the multiple sampling networks with the sample data set and assigning the multiple sampling networks trained as the multiple standby networks includes: assigning the first level network as a present first sampling network, and assigning each of at least one second level work and the third level network as a present second sampling network; processing the multiple pieces of input data based on the present first sampling network to determine a first output data corresponding to each input data, and determining a first loss value based on the first output data corresponding to each input data and a label corresponding to each input data; for each of multiple present second sampling networks: processing the multiple pieces of input data based on the present second sampling network to determine a second output data corresponding to each input data, and determining a second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data; determining the present first sampling network and the present second sampling network as the standby network, when the first loss value is less than or equal to a first preset threshold and the second loss value is less than or equal to a second preset threshold.

[11] In some possible embodiments, when the first loss value is less than or equal to the first preset threshold and the second loss value is less than or equal to the second preset threshold, the method further includes: before determining the present first sampling network and the present second sampling deep learning network as the standby network, back propagating based on the first loss value when the first loss value is greater than the first preset threshold, performing parameter update on the present first sampling network to obtain

an updated first sampling network, and determining the updated first sampling network as the present first sampling network; back propagating based on the second loss value when the second loss value is greater than the second preset threshold, performing parameter update on the present second sampling network to obtain an updated second sampling network, and determining the updated second sampling network as the present second sampling network; repeating the following: processing the multiple pieces of input data based on the present first sampling network to determine the first output data corresponding to each input data, and determining the first loss value based on the first output data corresponding to each input data and the label corresponding to each input data; and processing the multiple pieces of input data based on the present second sampling network to determine the second output data corresponding to each input data, and determining the second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data.

[12] In some possible embodiments, the original network includes one main branch and at least one slave branch, the main branch includes multiple network modules, and network modules in each slave branch includes part of the network modules in the main branch. If the original network has multiple slave branches, each of the multiple branches shares at least one network module.

[13] In some possible embodiments, each network module includes at least one convolutional module, and each convolutional module includes one convolutional layer, one batch normalization (BN) layer, and one activation layer. The convolutional layer, the BN layer, and the activation layer are connected in series. The convolutional layer is a depthwise convolutional layer, a pointwise convolutional layer, or a two-dimensional convolutional layer.

[14] In some possible embodiments, the device resource information of the original device at least includes: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the original device, the device resource information of the target device at least includes: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the target device.

[15] In some possible embodiments, determining the original network from the trained network set according to the target event includes: determining the original network from the trained network set according to the target event; determining a task identifier and an object identifier of each network in the network set; determining a first match degree value between

the target task identifier and each task identifier; determining a second match degree value between the target object identifier and each object identifier; and determining the original network from the network set according to the first match degree value and the second match degree value.

[16] In some possible embodiments, dividing the target network from the original network according to the device resource information of the target device and the device resource information of the original device includes: adjusting an output resolution, an output channel quantity, and a convolutional module quantity of each network module in the original network according to the device resource information of the target device and the device resource information of the original device, with a preset search rule; determining the original network adjusted as the original network.

[17] In some possible embodiments, an apparatus for determining a deep learning network is provided. The apparatus includes a target event determining unit, an original deep learning network determining unit, a resource information obtaining unit, and a target network dividing unit. The target event determining unit is configured to determine a target event to be processed by a target device. The original deep learning network determining unit is configured to determine an original network from a trained network set according to the target event, the original network being a network in the network set which matches the target event. The resource information obtaining unit is configured to obtain device resource information of an original device that trains the original network and device resource information of the target device. The target network dividing unit is configured to divide a target network from the original network according to the device resource information of the target device and the device resource information of the original device, the target network being deployed on the target device to process the target event, the target network and the original network are the same in the number of network modules and a connection structure between network modules.

[18] In some possible embodiments, the apparatus further includes a network training unit. The network training unit includes a data obtaining sub-unit, a network constructing sub-unit, and a network training sub-unit. The data obtaining sub-unit is configured to obtain a sample data set, the sample data set includes multiple pieces of input data and labels each corresponding to one piece of input data. The network constructing sub-unit is configured to construct a preset learning network, determining multiple sampling networks from the preset learning network, and set an initial parameter for the sampling network. The network training sub-unit is configured to train the multiple sampling networks with the sample data set and

assign the multiple sampling networks trained as multiple standby networks, and determine the original network from the multiple standby networks.

[19] In some possible embodiments, the multiple sampling networks include a first level network, at least one second level network, and a third level network, the first level network, the at least one second level network, and the third level network are same in the number of network modules and a connection structure between network modules. Each network module in the first level network having an output resolution greater than a corresponding network module in the second level network, and each network module in the second level network having an output resolution greater than a corresponding network module in the third level network. Each network module in the first level network having an output channel quantity greater than a corresponding network module in the second level network, and each network module in the second level network having an output channel quantity greater than a corresponding network module in the third level network. Each network module in the first level network having a convolution module quantity greater than a corresponding network module in the second level network, and each network module in the second level network having a convolution module quantity greater than a corresponding network module in the third level network.

[20] In some possible embodiments, the network training sub-unit is configured to: assign each of the multiple sampling networks as a present sampling network; process the multiple pieces of input data based on the present sampling network to determine an output data corresponding to each of the multiple pieces of input data; determine a loss value based on the output data and a label corresponding to the input data; determine the present sampling network as a standby network when the loss value is less than or equal to a preset threshold.

[21] In some possible embodiments, the network training sub-unit is configured to: back propagate based on the loss value when the loss value is greater than the preset threshold, perform parameter update on the present sampling network to obtain an updated sampling network, and determine the updated sampling network as the present sampling network; repeat the following: processing the multiple pieces of input data based on the present sampling network to determine the output data corresponding to each of the multiple pieces of input data, and determining the loss value based on the output data and the label corresponding to the input data.

[22] In some possible embodiments, the network train sub-unit is configured to: assign the first level network as a present first sampling network, and assign each of at least one second level work and the third level network as a present second sampling network; process

the multiple pieces of input data based on the present first sampling network to determine a first output data corresponding to each input data, and determine a first loss value based on the first output data corresponding to each input data and a label corresponding to each input data; for each of multiple present second sampling networks: process the multiple pieces of input data based on the present second sampling network to determine a second output data corresponding to each input data, and determine a second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data; determine the present first sampling network and the present second sampling network as the standby network, when the first loss value is less than or equal to a first preset threshold and the second loss value is less than or equal to a second preset threshold.

[23] In some possible embodiments, the network training sub-unit is configured to: back propagate based on the first loss value when the first loss value is greater than the first preset threshold, perform parameter update on the present first sampling network to obtain an updated first sampling network, and determine the updated first sampling network as the present first sampling network; back propagate based on the second loss value when the second loss value is greater than the second preset threshold, perform parameter update on the present second sampling network to obtain an updated second sampling network, and determine the updated second sampling network as the present second sampling network; repeat the following: processing the multiple pieces of input data based on the present first sampling network to determine the first output data corresponding to each input data, and determining the first loss value based on the first output data corresponding to each input data and the label corresponding to each input data; processing the multiple pieces of input data based on the present second sampling network to determine the second output data corresponding to each input data, and determining the second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data.

[24] In some possible embodiments, the original network includes one main branch and at least one slave branch, the main branch includes multiple network modules, and network modules in each slave branch includes part of the network modules in the main branch. If the original network has multiple slave branches, each of the multiple branches shares at least one network module.

[25] In some possible embodiments, each network module includes at least one convolutional module, and each convolutional module includes one convolutional layer, one batch normalization (BN) layer, and one activation layer. The convolutional layer, the BN

layer, and the activation layer are connected in series. The convolutional layer is a depthwise convolutional layer, a pointwise convolutional layer, or a two-dimensional convolutional layer.

[26] In some possible embodiments, the device resource information of the original device at least includes: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the original device, the device resource information of the target device at least includes: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the target device.

[27] In some possible embodiments, the original network determining unit is configured to: decompose the target event to determine a target object identifier and a target task identifier of the target event; determine a task identifier and an object identifier of each network in the network set; determine a first match degree value between the target task identifier and each task identifier; determine a second match degree value between the target object identifier and each object identifier; determine the original network from the network set according to the first match degree value and the second match degree value.

[28] In some possible embodiments, the target network dividing unit is configured to: adjust an output resolution, the number of output channels, and the number of convolutional modules of each network module in the original network according to the device resource information of the target device and the device resource information of the original device, with a preset search rule; and determine the original network adjusted as the original network.

[29] In some possible embodiments, an electronic device is provided. The electronic device includes at least one processor and a memory in communication with the at least one processor. The memory stores instructions which are executable with the at least one processor, the processor is configured to execute the instructions stored in the memory to perform the method of the second aspect of the disclosure.

[30] In some possible embodiments, a computer readable storage medium is provided. The computer storage medium stores at least one instruction or at least one program, the at least one instruction or the at least one program are uploaded and executed by a processor to perform the method of the second aspect of the disclosure.

[31] In some possible embodiments, a computer program product including instructions is provided. When the instructions are running on a computer, are operable with the computer to perform the method of the second aspect of the disclosure.

[32] In embodiments of the disclosure, a target event to be processed by a target device

is determined, an original network is determined from a trained network set according to the target event, device resource information of an original device that trains the original network and device resource information of the target device are obtained; a target network is divided from the original network according to the device resource information of the target device and the device resource information of the original device; the target network is to be deployed on the target device to process the target event, the target network and the original network are same in the number of network modules and a connection structure between network modules. As such, by determining the target network from the original network, researchers do not need to train the network for each device, which can save time spent on network design and training and greatly improve work efficiency.

[33] It should be understood that the above general description and the detailed description later are only exemplary and explanatory and do not limit the present disclosure.

[34] Other features and aspects of the present disclosure will become apparent from the following detailed description of exemplary embodiments with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[35] In order to more clearly explain the implementations of the disclosure or the technical solutions in the prior art, the following will briefly introduce the drawings required to be used in the implementations or the prior art description. It is obvious that the drawings in the following description are only some implementations of the disclosure. For those skilled in the art, other drawings can also be obtained from these drawings without paying creative labor.

[36] FIG. 1 is a structural diagram illustrating a system for determining a deep learning network according to embodiments of the disclosure.

[37] FIG. 2 is a flowchart illustrating a method for determining a deep learning network according to embodiments of the disclosure.

[38] FIG. 3 is a flowchart illustrating a method for determining an original network.

[39] FIG. 4 is a flowchart illustrating training to obtain an original network.

[40] FIG. 5 is a structural schematic diagram illustrating a preset learning network.

[41] FIG. 6 is a structural diagram illustrating a convolutional module.

[42] FIG. 7 is a structural diagram illustrating a convolutional module.

[43] FIG. 8 is a structural diagram illustrating a convolutional module.

[44] FIG. 9 is a module diagram illustrating an apparatus for determining a deep

learning network.

[45] FIG. 10 is a module diagram illustrating an electronic device according to embodiments of the disclosure.

[46] FIG. 11 is a module diagram illustrating another electronic device according to embodiments of the disclosure.

DETAILED DESCRIPTION

[47] Technical solutions in embodiments of the disclosure will be clearly and completely described below in combination with the accompanying drawings of the disclosure. Obviously, the described embodiments are only part rather than all of the embodiments of the disclosure. Based on the embodiments provided herein, all other embodiments obtained by those skilled in the art without creative work belong to the protection scope of the application.

[48] It should be noted that the terms “first” and “second” in the description and claims of the disclosure and the above drawings are used to distinguish similar objects, and does not necessarily used to describe a specific order or sequence. It should be understood that the data thus used can be interchanged where appropriate so that the embodiments of the present disclosure described herein can be implemented in an order other than those illustrated or described herein. In addition, the terms “include” and “have” and any variations thereof are intended to cover non-exclusive inclusion. For example, a process, method, system, product or server containing a series of steps or units need not be limited to those steps or units clearly listed, but may include those not clearly listed or inherent for these processes, methods, products or equipment.

[49] Various exemplary embodiments, features, and aspects of the present disclosure will be described in detail below with reference to the accompanying drawings. The same reference numerals in the drawings represent elements with the same or similar functions. Although various aspects of the embodiments are shown in the drawings, it is not necessary to draw the drawings to scale unless specifically noted.

[50] The special word “exemplary” here means “as an example, embodiment or illustrative”. Any embodiment described herein as “exemplary” need not be interpreted as superior or better than other embodiments.

[51] The term “and/or” used herein only refers to an association relationship describing associated objects, representing that there can be three relationships, for example, “a and / or B” can mean that there are three situations: a alone, both a and B, and B alone. The term “at

least one” herein means any one of a plurality or any combination of at least two of a plurality, for example, “including at least one of A, B and C” may mean including any one or more elements selected from the set composed of A, B and C.

[52] In addition, in order to better illustrate the present disclosure, many specific details are given in the following specific embodiments. Those skilled in the art should understand that the present disclosure can also be implemented without some specific detail. In some examples, methods, means, elements and circuits well known to those skilled in the art are not described in detail in order to highlight the subject matter of the present disclosure.

[53] FIG. 1 is a structural diagram illustrating a system for determining a deep learning network according to embodiments of the disclosure. As illustrated in FIG. 1, a first device 10 and a second device 20 are included.

[54] Optionally, the first device 10 can collect various deep learning networks from other devices, and put all collected deep learning networks into a set, and store the set into a storage area, so as to determine an original network related to a target event later, and then determine a target network. Optionally, the other devices may include the first device per se. The original network may be a deep learning network(s) in all the collected various deep learning networks that associated with the target event.

[55] The first device 10 may include, but is not limited to, smart phones, desktop computers, tablet computers, laptops, smart speakers, digital assistants, augmented reality (AR) / virtual reality (VR) devices, smart wearable devices and other types of electronic devices. The first device 10 may be a server. Optionally, an operating system running on the electronic device or the server may include but is not limited to Android system, IOS system, Linux, Windows, Unix, etc.

[56] The second device 20 is a target device that may need to deploy the target network, and is configured to receive a target network which is divided from the original network and deploy the target network on the device to process the target event. The second device 20 may include, but is not limited to, smart phones, desktop computers, tablet computers, laptops, smart speakers, digital assistants, AR/ VR devices, smart wearable devices and other types of electronic devices. The second device 20 may be a server. Optionally, an operating system running on the electronic device or the server may include but is not limited to Android system, IOS system, Linux, Windows, Unix, etc.

[57] In some possible embodiments, the first device 10 determines a target event to be processed by the target device (the second device 20), and determine an original network from a trained network set according to the target event. The original network is a network in

the network set that matches the target event. The first device 10 obtains device resource information of an original device that trains the original network and device resource information of a target device, and divides the target network from the original network according to the device resource information of the target device and the device resource information of the original device. The target network is to be deployed on the target device (the second device 20) to process the target event. The target network and the original network are the same in the number of network modules and a connection structure between network modules.

[58] In one possible embodiment, the first device 10 and the second device 20 are connected through a wired link or a wireless link.

[59] The method for determining a deep learning network according to embodiments of the disclosure will be described, the method is performed by a server for example. The method can be implemented by calling a computer readable instruction stored in a memory by a processor.

[60] FIG. 2 is a flowchart illustrating a method for determining a deep learning network according to embodiments of the disclosure. As illustrated in FIG.2, the method includes the following.

[61] At S201, a target event to be processed by a target device is determined.

[62] In embodiments of the disclosure, the server deploys on the target device a network capable of processing the target event. Since some networks trained in advance have been stored, the server can select from the networks directly an original network that can process the target event. The server adjusts the original network based on device resource information of the target device to obtain the target network. Based on the above, the target event to be processed by the target device needs to be determined first.

[63] In some possible embodiments, the target event can be various, for example, image recognition, image segmentation, (equipment, machinery) fault recognition, obstacle recognition, trajectory prediction, article scoring, and so on. For ease of understanding, image segmentation will be described as the target event below. For other events, please refer to image segmentation event, and specific contents thereof will not be repeated.

[64] In some possible embodiments, the target device can be the server. That is to say, the server determines the target event to be processed for the server per se. In some other possible embodiments, the target device is device other than the server. Specifically, the server can receive a network requirement command from the target device, the network requirement command carries an identifier (ID) of the target event, and the network

requirement command indicates that the server determines the target event according to the ID of the target event.

[65] At S202, the original network is determined from a trained network set according to the target event, the original network is a network in the network set which matches the target event.

[66] In embodiments of the disclosure, optionally, the trained network set can be stored in the server in advance. Once the target event is determined, the original network can be determined from the trained network set directly. Optionally, the server can obtain multiple trained networks from devices connected with the server after the target event is determined, and determine the original network therefrom, where the original network is a network in the network set that matches the target event. In other words, the original network is a network that can process the target event. For example, if the target event is image segmentation, the original network determined is a network that has processed the image segmentation event.

[67] However, in some possible embodiments, there is no network in the network set that corresponds to the target event, that is, for example, there is no network in the network set that has processed or handled the image segmentation event. In this case, FIG. 3 is a flowchart illustrating a method for determining an original network according to embodiments of the disclosure. As illustrated in FIG. 3, the method includes the following.

[68] At S201, the target event is decomposed to determine a target object ID and a target task ID of the target event.

[69] In some possible embodiments, the server can recognize the target event to obtain the target object ID and the target task ID of the target event.

[70] In some other possible embodiments, the network requirement command carries the ID of the target event, and the ID of the target event includes the target object ID and the target task ID. After receiving the ID of the target event, the server can decompose the ID of the target event to obtain the target task ID and the target object ID. The target object ID is for indicating an object to be processed by the target network, which is finally determined by the server, and the target task ID is for indicating the manner in which the object is processed by the target network, which is finally determined by the server.

[71] For example, the “image recognition” event can be decomposed into: an object to be processed is an “image”, and the manner in which the object is processed is “recognition”, therefore, the task ID of “image recognition” can be the ID corresponding to “recognition”, and the object ID can be the ID corresponding to “image”. Still another example, the “image segmentation” event can be decomposed into: an object to be processed is an “image”, and

the manner in which the object is processed is “segmentation”, therefore, the task ID of “image segmentation” can be the ID corresponding to “segmentation”, and the object ID can be the ID corresponding to “image”. Still another example, “article scoring” event can be decomposed into: an object to be processed is an “article”, and the manner in which the object is processed is “scoring”, therefore, the task ID of “article scoring” can be the ID corresponding to “scoring”, and the object ID can be the ID corresponding to “article”. In some possible embodiments, the task ID and the image ID are unique IDs, for example, in the object ID, the ID corresponding to “image” is A001, the ID corresponding to “Article” is A002....., in the task ID, the ID corresponding to “recognition” is B001, the ID corresponding to “segmentation” is B002, and the ID corresponding to “scoring” is B003, etc.

[72] Based on the above examples, at S2021, if the target event is “image segmentation”, the server can determine that the target task ID of the target event is B002 and the target object ID of the target event is A001.

[73] At S2022, the task ID and object ID for each network in the network set is obtained.

[74] In some possible embodiments, when the server obtaining various networks, each network may carry its own task ID and object ID. In some other possible embodiments, when the server obtaining various networks, each network may carry the ID of an event, and the server can decompose the ID of the event to obtain the task ID and the object ID of each event.

[75] At S2023, a first match degree value between the target task ID and each task ID is determined.

[76] At S2024, a second match degree value between the target object ID and each object ID is determined.

[77] To be continued, the target event is image segmentation. In some possible embodiments, the target task ID is compared with the task ID of each network in the network set, if they are the same, then the first match degree value is 1, otherwise, the first match degree value is 0. Similarly, the target object ID is compared with the object ID of each network in the network set, if they are the same, then the second match degree value is 1, otherwise, the second match degree value is 0.

[78] However, the above matching manner is too absolute. For example, if the target object ID is “image A001” and the object ID is “picture A101”, according to the above method, the second match degree value is 0 and this is not reasonable and may even lead to the failure of the whole scheme. Therefore, in some possible embodiments, take the target

object ID as an example, similar extended object ID can be determined according to the target object ID, for example, similar extended object ID “picture A101”, “photo A012” and so on can be determined according to “image A001”. Thereafter, the target task ID and the extended object ID can each be compared with the object ID to obtain the second match degree value between the target object ID and the object ID. Specifically, if the manner in which the first match degree value between the target task ID and the task ID is determined can make reference to that of the object ID, extended task IDs corresponding to the target task ID can be determined first.

[79] At S2025, the original network is determined from the network set according to the first match degree value and the second match degree value.

[80] In some possible embodiments, a network with both the first match degree value and the second match degree value of 1 can be determined as the original network.

[81] Embodiments of the disclosure further includes a procedure of training to obtain the original network. In one embodiment, the original network includes but is not limited to deep learning networks such as a convolutional neural network, a recurrent neural network or a recursive neural network. Deep learning network can be considered as a form of machine learning (ML). ML is a multi-domain interdisciplinary subject, involving probability theory, statistics, approximation theory, convex analysis, algorithm complexity theory and so on. ML focuses on how computers simulate or realize human learning behavior, so as to obtain new knowledge or skills, reorganize the existing knowledge structure and continuously improve its performance. ML is the core of artificial intelligence and the fundamental way to make computers intelligent, it is applied in all fields of artificial intelligence. ML and deep learning usually include artificial neural network, confidence network, reinforcement learning, transfer learning, inductive learning, type teaching learning and so on. ML can be divided into supervised machine learning, unsupervised machine learning and semi supervised machine learning.

[82] In one embodiment, taking the convolution neural network as an example, FIG. 4 is a flowchart illustrating a procedure of training to obtain the original network. As illustrated in FIG. 4, the method includes the following.

[83] At S401, a sample data set is obtained, the sample data set includes multiple pieces of input data and labels each corresponding to one piece of input data.

[84] If an image segmentation event is previously processed by the original network, before training the original network, two preparations need to be made: i) construct a preset learning network; ii) collecting a sample data set. Take image segmentation as an example,

the server needs to obtain an image.

[85] In some possible embodiments, the server can download images from an online image library or obtain images through other electronic devices, for example, the server can intercept or capture images from the video recorded by a vehicle's camera. For example, the image is intercepted from a video screen recorded with a frame rate of 30 frames by the vehicle camera, one image is intercepted every 10 frames, 3 images are intercepted every second, and the intercepted images are transmitted to the server. In some possible embodiments, the server can perform pre-processing on the image according to actual needs, the pre-processing includes image size changing, gray value processing, Gaussian filter processing or gamma correction, etc. Optionally, in addition to obtaining the image, the server still has to obtain the label corresponding to the image, for example, objects (people, dogs, plants, furniture, etc.) contained in the image, furthermore, the label may further include the location of the object in the image.

[86] At S402, a preset learning network is constructed, multiple sampling networks are determined from the preset learning network, and an initial parameter(s) is set for each sampling network.

[87] FIG. 5 is a structural schematic diagram illustrating a preset learning network according to embodiments of the disclosure. As illustrated in FIG. 5, the preset learning network in S402 can be referred to as Hypernetwork or search space, it can be the network illustrated in FIG. 5(a) which includes only one branch, or can be the network illustrated in FIG. 5(b) which includes two branches, or can be the network illustrated in FIG. 5(c) which includes five branches. The number of branches in the network can be set according to actual needs, which is not limited herein.

[88] Considering that the network should meet the requirements of real-time network processing and small capacity at the same time, embodiments of the disclosure will be described with a network including two branches. The two-branch network illustrated in FIG. 5(b) includes a main branch on the top and a slave branch on the bottom. The main branch includes a first network module, a second network module, a third network module, a fourth network module, a fifth network module, and a sixth network module connected in series from the start to the end. The main branch can capture high-level semantic information (such as type information, correlation between objects, and context information) in an image segmentation event. The slave branch includes the first network module, the second network module, and the third network module connected in series from the start to the end. The slave branch can capture underlying semantic information (such as contour information, size

information, color information, etc.) in the image segmentation event. The first network module, the second network module, and the third network module are common to the main branch and the slave branch.

[89] In embodiments of the disclosure, for each network module in the two-branch network, resolution, type of the convolutional module, the number of channels and/or the number of convolutional modules can be preset. Since the multiple sampling networks need to be determined from the preset learning network, the resolution, the type of the convolutional module, the number of channels and/or the number of convolutional module can be a range, as illustrated in Table 1.

[90] Table 1: Network structure unit

Slabele	Resolution	Type of the convolutional module	The number of convolutional modules	The number of channels
First network module	896*1792-1536*3072	2D convolution	1	24-40
Second network module	448*896-768*1536	Depthwise separable convolution	1	32-64
Third network module	224*448-384*768	Depthwise separable convolution	1	48-72
Fourth network module	112*224-192*384	Dilated depthwise separable convolution	2-4	48-72
Fifth network module	56*112-96*192	Dilated depthwise separable convolution	2-4	72-128
Sixth network module	28*56-48*96	Dilated depthwise separable convolution	2-4	96-160

[91] In the above Table, line 2 represents that the range of the output resolution of the first network module is 896*1792-1536*3072, the type of the convolutional module is 2D convolution, the number of convolutional modules is 1, and the range of the number of output channels is 24-40.

[92] Line 3 represents that the range of the output resolution of the second network module is 448*896-768*1536, the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the range of the number of output channels is 32-64.

[93] Line 4 represents that the range of the output resolution of the third network module is 224*448-384*768, the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the range of the number of output channels is 48-72.

[94] Line 5 represents that the range of the output resolution of the fourth network

module is $112*224-192*384$, the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2-4, and the range of the number of output channels is 48-72.

[95] Line 6 represents that the range of the output resolution of the fifth network module is $56*112-96*192$, the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2-4, and the range of the number of output channels is 72-128.

[96] Line 7 represents that the range of the output resolution of the sixth network module is $28*56-48*96$, the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2-4, and the range of the number of output channels is 96-160.

[97] At S402, determining the multiple sampling networks from preset learning networks can include obtaining numerous sampling networks according to different resolutions of network modules, different types of convolutional modules, different numbers of channels, and/or different numbers of convolutional modules. Alternatively, the multiple sampling networks can be determined considering computing ability of the device training the sampling network or practical application.

[98] In some possible implementations, the multiple sampling networks may include sampling networks with randomly determined resolution, the number of the channels, and/or the number of convolutional modules. Alternatively, the multiple sampling networks may include a first level network (or referred to as a maximum model), at least one second level network and a third level network (or referred to as a minimum model). The number of network modules in each of the first level network, the at least one second level network and the third level network is the same. For example, as illustrated in FIG. 5(b), the main branch has six network modules, the slave branch has three network modules, and the connection structure between the network modules are the same.

[99] For example, the first level network can include a network in which the network modules have the maximum output resolution, the maximum number of output channels, and the maximum number of convolutional modules. Taking the data in Table 1 as an example, in the first level network, for the first network module, the output resolution is $1536*3072$, the type of the convolutional module is 2D convolution, the number of convolutional modules is 1, and the number of output channels is 40; for the second network module, the output resolution is $768*1536$, the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is

64; for the third network module, the output resolution is 384×768 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 72; for the fourth network module, the output resolution is 192×384 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 4, and the number of output channels is 72; for the fifth network module, the output resolution is 96×192 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 4, and the number of output channels is 128; for the sixth network module, the output resolution is 48×96 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 4, and the number of output channels is 160.

[100] The third level network can include a network in which the network modules have the minimum output resolution, the minimum number of output channels, and the minimum number of convolutional modules. Taking the data in Table 1 as an example, in the third level network, for the first network module, the output resolution is 896×1792 , the type of the convolutional module is 2D convolution, the number of convolutional modules is 1, and the number of output channels is 24; for the second network module, the output resolution is 448×896 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 32; for the third network module, the output resolution is 224×448 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 48; for the fourth network module, the output resolution is 112×224 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2, and the number of output channels is 48; for the fifth network module, the output resolution is 56×112 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2, and the number of output channels is 72; for the sixth network module, the output resolution is 28×56 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2, and the number of output channels is 96.

[101] Each network module in the second level network is arranged between the first level network and the third level network. The output resolution of each network module in the first level network is greater than that of a corresponding network module in the second level network. The output resolution of each network module in the second level network is greater than that of a corresponding network module in the third level network. The number

of output channels of each network module in the first level network is greater than that of a corresponding network module in the second level network. The number of output channels of each network module in the second level network is greater than that of a corresponding network module in the third level network. The number of convolutional modules of each network module in the first level network is greater than that of a corresponding network module in the second level network. The number of convolutional modules of each network module in the second level network is greater than that of a corresponding network module in the third level network.

[102] For example, there are four sampling networks, then there are two second level networks. The second level network 1 can be: for the first network module, the output resolution is 1236×2456 , the type of the convolutional module is 2D convolution, the number of convolutional modules is 1, and the number of output channels is 36; for the second network module, the output resolution is 668×1036 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 48; for the third network module, the output resolution is 364×668 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 56; for the fourth network module, the output resolution is 184×358 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 3, and the number of output channels is 68; for the fifth network module, the output resolution is 86×168 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 3, and the number of output channels is 108; for the sixth network module, the output resolution is 36×72 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 3, and the number of output channels is 156. The second level network 2 can be: for the first network module, the output resolution is 1036×1856 , the type of the convolutional module is 2D convolution, the number of convolutional modules is 1, and the number of output channels is 28; for the second network module, the output resolution is 548×936 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 32; for the third network module, the output resolution is 264×548 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 50; for the fourth network module, the output resolution is 124×256 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2, and the

number of output channels is 60; for the fifth network module, the output resolution is 66×128 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 3, and the number of output channels is 108; for the sixth network module, the output resolution is 32×72 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 3, and the number of output channels is 114.

[103] In some possible implementations, the main computing of the convolution module is at a convolutional layer, in order to reduce the amount of computing, the server can provide different convolutional modules. The amount of computing of the convolutional layer can be expressed as $C_{out} \times C_{in} \times k \times k$, where C_{out} represents the number of output channels corresponding to a convolution kernel, C_{in} represents the number of input channels corresponding to the convolution kernel, and k represents the size of the convolution kernel. In order to reduce the amount and capacity of computing, the number of channels and/or the size of the convolution kernel can be reduced. Therefore, in some network modules with large amount of computing, 2D convolution can be replaced with depthwise separable convolution having a small amount of computing. This is because in the same computing environment, the amount of computing of depthwise separable convolution is smaller than that of 2D convolution. The depthwise separable convolution decompose a $C_{out} \times C_{in} \times k \times k$ convolution kernel into two dimensions, that is, depthwise convolution ($1 \times 1 \times k \times k$) and pointwise convolution ($C_{out} \times C_{in} \times 1 \times 1$), therefore, the depthwise separable convolution can be divided into two convolution layers, that is, depthwise convolutional layer and pointwise convolutional layer.

[104] In some possible implementations, according to the type of the convolutional module, the convolutional module may include at least one convolutional layer. Optionally, if the convolutional module is 2D convolution, the convolutional module may include only one 2D convolutional layer. If the convolutional module is a depthwise separable convolution, the convolutional module may include two convolutional layers connected in series, that is, a depthwise convolutional layer and a pointwise convolutional layer. If the convolutional module is a dilated depthwise separable convolution, the convolutional module may include three convolutional layers connected in series, that is, a pointwise convolutional layer, a depthwise convolutional layer, and a pointwise convolutional layer.

[105] FIG. 6 is a structural diagram illustrating a convolutional module. As illustrated in FIG. 6, the convolutional module includes a depthwise separable convolutional module on

the left and a dilated depthwise separable convolutional module. The depthwise separable convolutional module includes a depthwise convolutional layer with a 3*3 convolutional kernel and a pointwise convolutional layer with a 1*1 convolutional kernel. The dilated depthwise separable convolutional module includes a pointwise convolutional layer with a 1*1 convolutional kernel, a depthwise convolutional layer with a 3*3 convolutional kernel, and a pointwise convolutional layer with a 1*1 convolutional kernel. In actual applications, data is often not linearly separable. Therefore, in order to introduce a nonlinear factor, an activation layer (ReLU) can be introduced subsequent to the convolutional layer.

Furthermore, in order to standardize, translate, and scale the data output by the convolution layer and prevent the data from reaching saturation, a batch normalized (BN) layer can be arranged between the convolution layer and the activation layer, as such, the structure of the convolutional module illustrated in FIG. 7 can be obtained. In addition, the residual structure in the convolution module shown in Fig. 8 can be used to solve the network degradation problem in dilated depthwise separable convolution.

[106] In some possible embodiments, before training the sampling networks, an initial parameter(s) has to be set for each sampling network. For example, for all convolutional networks, the server can use `kaiming_init`, Xavier, and so on for initialization. For some batch normalized (BN) layers, a scaling parameter γ can be initialized to 1, and translation parameter β is initialized to 0. Specifically, for a residual structure in dilated depthwise separable convolution, the scaling parameter γ of the last batch normalized (BN) layer can be initialized to 0, as such, the output of all residual modules before residual connection can be all 0 vectors, which can improve the stability of training.

[107] At S403, the multiple sampling networks are trained with the sample data set and the multiple sampling networks trained are assigned as multiple standby networks, and the original network is determined from the multiple standby networks.

[108] In some possible implementations, the training can include: forward propagating each sampling network after setting the initial parameters as above, to obtain the output under the specified input, thereafter, the gradient of parameters and input can be obtained through back propagation, and then gradient processing (such as limiting gradient range) and updating can be carried out. Finally, multiple trained standby networks can be obtained.

[109] In one possible implementation, each sampling network is trained in the same way. Specifically, for each of multiple sampling networks: the sampling network is assigned as a present sampling network, and multiple pieces of input data is processed based on the

present sampling network and output data corresponding to each input data is determined. Thereafter, the server determines a loss value based on the output data and a label corresponding to the input data. When the loss value is greater than a preset threshold, back propagation is carried out based on the loss value, parameter update is performed on the present sampling network to obtain an updated sampling network, and the updated sampling network is determined as the present sampling network. Repeat the following: the multiple pieces of input data is processed based on the present sampling network and output data corresponding to each input data is determined; the loss value is determined based on the output data and the label corresponding to the input data. Until the loss value is less than or equal to the preset threshold, the present sampling network is determined as a standby network. In this way, it seems that each sampling network is trained as the present sampling network, but actually, all sampling networks are trained at the same time. But the server can train each sampling network at different times either, as such, the server can be prevented from affecting other processes due to the significant consumption of computing capacity, and will not result in server downtime.

[110] In another possible implementation, to save the computing resource of the server and further reduce the burden of the server, the server can assign the first level network as a “teacher” network directly or according to preset settings, and assign the second level network and the third level network as “student” networks. As such, in network training, the performance of the first level network can be directly migrated to the second level network and the third level network.

[111] Specifically, the server can assign the first level network as a present first sampling network, assign each of the at least one second level work and the third level network as a present second sampling network. The multiple pieces of input data is processed based on the present first sampling network to determine a first output data corresponding to each input data, and a first loss value is determined based on the first output data corresponding to each input data and a label corresponding to each input data. For each of multiple present second sampling networks: the multiple pieces of input data is processed based on the present second sampling network to determine a second output data corresponding to each input data, and a second loss value is determined based on the second output data corresponding to each input data and the first output data corresponding to each input data. When the first loss value is greater than a first preset threshold, back propagation is performed based on the first loss value, parameter update is performed on the present first sampling network to obtain an updated first sampling network, and the updated first sampling

network is determined as the present first sampling network. When the second loss value is greater than a second preset threshold, back propagation is performed based on the second loss value, parameter update is performed on the present second sampling network to obtain an updated second sampling network, and the updated second sampling network is determined as the present second sampling network. The following is repeated: the multiple pieces of input data is processed based on the present first sampling network to determine a first output data corresponding to each input data, and the first loss value is determined based on the first output data corresponding to each input data and the label corresponding to each input data; the multiple pieces of input data is processed based on the present second sampling network to determine the second output data corresponding to each input data, and the second loss value is determined based on the second output data corresponding to each input data and the first output data corresponding to each input data. The present first sampling network and the present second sampling network are determined as standby networks only when the first loss value is less than or equal to the first preset threshold and the second loss value is less than or equal to the second preset threshold.

[112] The first output data of the present first sampling network can be a soft label. In terms of the image segmentation event, the first output data can be multiple segmented objects and the probability value of each object, not just the hard label of the object. As such, when training at the present second sampling network, the soft label can be used as a true value to be compared with the second output data to determine the loss value. As such, the useful information of the larger and more complex large model (the first level network) with good performance can be extracted and migrated to a smaller network (the second level network and the third level network). In this way, the learned small network can have the performance close to that of the large network, and greatly save computing resources. Optionally, the large model (the first level network) requires the server to regularize it, that is, a regularization layer needs to be added to the network model. This is because only the largest model can directly contact the label, and other small models supervise the training through the output of the large model, that is, soft label distillation.

[113] In some possible implementations, during training, the large model usually reaches the accuracy peak first and then starts over fitting, while the small model is still in the state of under fitting. Therefore, the server can solve this problem by modifying the learning rate according to preset rules. For example, the server can first use an exponential decay method to reduce the learning rate, and keep the learning rate constant when the learning rate reaches an initial set value. The advantage of this is that using a slightly larger learning rate

can make the small model learn faster. When the large model reaches the accuracy peak, giving a constant learning rate will make the gradient oscillate, so as to alleviate the over fitting phenomenon.

[114] With aid of training manners including those described above, the server can determine multiple standby networks. In one embodiment, each standby network can be treated as an original network that can process the image segmentation event. Alternatively, the performance of each network can be determined. For example, the accuracy of each network and the computing resources to be used can be determined through experimental samples, and one or more original networks can be determined according to the performance of each network.

[115] Since the original network and a corresponding sampling network (in other words, a Hyper network) are the same in structure, the original network includes one main branch and at least one slave branch. The main branch may include multiple network modules, and network modules in each slave branch include part of the network modules in the main branch. If the original network includes multiple slave branches, each slave branch at least shares one network module. Furthermore, each network module includes at least one convolutional module, and each convolutional module includes at least one convolutional layer, a batch normalized (BN) layer, and an activation layer. The convolutional layer, the batch normalized (BN) layer, and the activation layer are connected in series. The convolutional layer can be a depthwise convolutional layer, a pointwise convolutional layer, or a 2D convolutional layer.

[116] At S203, device resource information of an original device training the original network and device resource information of a target device are obtained. The device resource information of the original device at least comprises: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the original device. The device resource information of the target device at least comprises: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the target device.

[117] Optionally, the device resource information of the original device can be provided by the original network and thus can be obtained by the server from the original network directly. Alternatively, the device resource information of the original device can be obtained from a device resource information storage area of the original device and then transmitted to the server.

[118] Optionally, the server can obtain the target event to be processed by the target device, and obtain the device resource information of the target device from the device resource information storage area of the target device. The above two steps can be implemented at the same time or implemented one after the other. As to which of the two steps comes first and which comes later, it is not limited herein, as long as it is appropriate.

[119] At S204, a target network is divided from the original network according to the device resource information of the target device and the device resource information of the original device, the target network is to be deployed on the target device to process the target event, the target network and the original network are the same in the number of network modules and a connection structure between network modules.

[120] In some possible implementations, the server can adjust the output resolution, the number of output channels, and the number of convolutional modules of each network module in the original network according to the device resource information of the target device and the device resource information of the original device with a preset searching rule, and the adjusted original network is assigned as the target network. The preset searching rule may include random search algorithm, genetic search algorithm, differentiable network structure parameter search, reinforcement learning based search algorithm, grid search algorithm, and so on.

[121] For example, in the original network, for the first network module, the output resolution is 896×1792 , the type of the convolutional module is 2D convolution, the number of convolutional modules is 1, and the number of output channels is 24; for the second network module, the output resolution is 448×896 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 32; for the third network module, the output resolution is 224×448 , the type of the convolutional module is depthwise separable convolution, the number of convolutional modules is 1, and the number of output channels is 48; for the fourth network module, the output resolution is 112×224 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2, and the number of output channels is 48; for the fifth network module, the output resolution is 56×112 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 2, and the number of output channels is 72; for the sixth network module, the output resolution is 28×56 , the type of the convolutional module is dilated depthwise separable convolution, the number of convolutional modules is 3, and the number of output channels is 96. Each device resource information of the target device is half of each

device resource information of the original device. For example, the original device is a desktop and the target device is a mobile phone, for each network module, the output resolution, the number of output channels, and the number of convolutional modules can be reduced. Or, the device resource information of the target device is basically similar to the device resource information of the original device except that there is one device resource of the target device greater than the same device resource of the original device, therefore, for some network modules, the output resolution, the number of output channels, and the number of convolutional modules can be increased.

[122] In some possible embodiments, to be on the safe side and making the target network more suitable for the target device, after the target network is deployed on the target device, the target network can run on the target device, so as to determine whether further adjustment on the output resolution, the number of output channels, and the number of convolutional modules are required based on the running result, so as to make sure the target network is more suitable for the target device.

[123] In some possible embodiments, in order to decouple training and search and obtain better performance of the target network, the process of training the original network and searching for the target network from the original network can be separated as described above. Alternatively, the process of training the original network and the process of searching for the target network from the original network can be intercross carried out. This can be determined according to actual needs and is not limited herein.

[124] To be summarized, embodiments of the disclosure provide a preset learning network (Hyper network, such as the two-branch network). After training, the target network suitable for various target devices can be obtained from the original network with a network structure search algorithm according to the device resource information obtained without training each target network, this can not only reduce the time spent by researchers in network design and training, but also make the trained original network more valuable.

[125] FIG. 9 is a module diagram illustrating an apparatus for determining a deep learning network. As illustrated in FIG. 9, the apparatus for determining a deep learning network includes a target event determining unit 901, an original network determining unit 902, a resource information obtaining unit 903, and a target network dividing unit 904.

[126] The target event determining unit 901 is configured to determine a target event to be processed by a target device. The original network determining unit 902 is configured to determine an original network from a trained network set according to the target event, the original network being a network in the network set which matches the target event. The

resource information obtaining unit 903 is configured to obtain device resource information of an original device that trains the original network and device resource information of the target device. The target network dividing unit 904 is configured to divide a target network from the original network according to the device resource information of the target device and the device resource information of the original device, the target network being deployed on the target device to process the target event, the target network and the original network are the same in the number of network modules and a connection structure between network modules.

[127] In some possible embodiments, the apparatus further includes a network training unit. The network training unit includes a data obtaining sub-unit, a network constructing sub-unit, and a network training sub-unit.

[128] The data obtaining sub-unit is configured to obtain a sample data set, the sample data set includes multiple pieces of input data and labels each corresponding to one piece of input data. The network constructing sub-unit is configured to construct a preset learning network, determining multiple sampling networks from the preset learning network, and set an initial parameter for the sampling network. The network training sub-unit is configured to train the multiple sampling networks with the sample data set and assign the multiple sampling networks trained as multiple standby networks, and determine the original network from the multiple standby networks.

[129] In some possible embodiments, the multiple sampling networks include a first level network, at least one second level network, and a third level network, the first level network, the at least one second level network, and the third level network are same in the number of network modules and a connection structure between network modules.

[130] Each network module in the first level network having an output resolution greater than a corresponding network module in the second level network, and each network module in the second level network having an output resolution greater than a corresponding network module in the third level network.

[131] Each network module in the first level network having an output channel quantity greater than a corresponding network module in the second level network, and each network module in the second level network having an output channel quantity greater than a corresponding network module in the third level network.

[132] Each network module in the first level network having a convolution module quantity greater than a corresponding network module in the second level network, and each network module in the second level network having a convolution module quantity greater

than a corresponding network module in the third level network.

[133] In some possible embodiments, the network training sub-unit is configured to: for each of the multiple sampling networks, assign the sampling network as a present sampling network; process the multiple pieces of input data based on the present sampling network to determine an output data corresponding to each of the multiple pieces of input data; determine a loss value based on the output data and a label corresponding to the input data; perform back propagation based on the loss value when the loss value is greater than the preset threshold, perform parameter update on the present sampling network to obtain an updated sampling network and assign the updated sampling network as the present sampling network; repeat the following: processing the multiple pieces of input data based on the present sampling network to determine an output data corresponding to each of the multiple pieces of input data and determining a loss value based on the output data and a label corresponding to the input data; determine the present sampling network as a standby network when the loss value is less than or equal to a preset threshold.

[134] In some possible embodiments, the network train sub-unit is configured to: assign the first level network as a present first sampling network, and assign each of at least one second level work and the third level network as a present second sampling network; process the multiple pieces of input data based on the present first sampling network to determine a first output data corresponding to each input data, and determine a first loss value based on the first output data corresponding to each input data and a label corresponding to each input data; for each of multiple present second sampling networks: process the multiple pieces of input data based on the present second sampling network to determine a second output data corresponding to each input data, and determine a second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data; perform back propagation based on the first loss value when the first loss value is greater than the first preset threshold, perform parameter update on the present first sampling network to obtain an updated first sampling network, and determine the updated first sampling network as the present first sampling network; perform back propagation based on the second loss value when the second loss value is greater than the second preset threshold, perform parameter update on the present second sampling network to obtain an updated second sampling network, and determine the updated second sampling network as the present second sampling network; repeat the following: processing the multiple pieces of input data based on the present first sampling network to determine the first output data corresponding to each input data, and determining the first loss value based on the first output data

corresponding to each input data and the label corresponding to each input data; processing the multiple pieces of input data based on the present second sampling network to determine the second output data corresponding to each input data, and determining the second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data; determine the present first sampling network and the present second sampling network as the standby network, when the first loss value is less than or equal to a first preset threshold and the second loss value is less than or equal to a second preset threshold.

[135] In some possible embodiments, the original network includes one main branch and at least one slave branch, the main branch includes multiple network modules, and network modules in each slave branch includes part of the network modules in the main branch. If the original network has multiple slave branches, each of the multiple branches shares at least one network module.

[136] In some possible embodiments, each network module includes at least one convolutional module, and each convolutional module includes one convolutional layer, one batch normalization (BN) layer, and one activation layer. The convolutional layer, the BN layer, and the activation layer are connected in series. The convolutional layer is a depthwise convolutional layer, a pointwise convolutional layer, or a two-dimensional convolutional layer.

[137] In some possible embodiments, the device resource information of the original device at least includes: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the original device, the device resource information of the target device at least includes: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the target device.

[138] In some possible embodiments, the original network determining unit is configured to: decompose the target event to determine a target object identifier and a target task identifier of the target event; determine a task identifier and an object identifier of each network in the network set; determine a first match degree value between the target task identifier and each task identifier; determine a second match degree value between the target object identifier and each object identifier; determine the original network from the network set according to the first match degree value and the second match degree value.

[139] In some possible embodiments, the target network dividing unit is configured to: adjust an output resolution, the number of output channels, and the number of convolutional

modules of each network module in the original network according to the device resource information of the target device and the device resource information of the original device, with a preset search rule; and determine the original network adjusted as the original network.

[140] In some implementations, functions or modules of the apparatus provided in embodiments of the disclosure can perform the method described in the above embodiments, implementation thereof can make reference to the above embodiments, which will not be repeated herein.

[141] Embodiments of the disclosure provide a computer readable storage medium. The computer storage medium stores at least one instruction or at least one program, the at least one instruction or the at least one program are uploaded and executed by a processor to perform the method of the second aspect of the disclosure. The computer readable storage medium can be a non-transitory computer readable storage medium.

[142] Embodiments of the disclosure provide an electronic device. The electronic device includes a processor and a memory storing instructions which are executable with the processor. The processor is configured to implement the above method.

[143] The electronic device can be a terminal, server, or other forms of devices.

[144] Embodiments of the disclosure provide a computer program product including instructions which, when running on a computer, are operable with the computer to perform the method for determining a deep learning network of the disclosure.

[145] FIG. 10 is a block diagram of an electronic device according to embodiments of the present disclosure. For example, the electronic device 1000 may be a mobile phone, a computer, a digital broadcasting terminal, a messaging device, a game console, a tablet device, a medical device, a fitness device, a personal digital assistant, etc.

[146] Referring to FIG. 10, the electronic device 1000 may include one or more of the following components: a processing component 1002, a memory 1004, a power supply component 1006, a multimedia component 1008, an audio component 1010, an input / output (I / O) interface 1012, a sensor component 1014, and a communication component 1016.

[147] The processing component 1002 typically controls the overall operation of the electronic device 1000, such as operations associated with display, telephone call, data communication, camera operation, and recording operation. The processing component 1002 may include one or more processors 1020 to execute instructions to complete all or part of the steps of the above method. In addition, the processing component 1002 may include one or more modules to facilitate interaction between the processing component 1002 and other components. For example, the processing component 1002 may include a multimedia module

to facilitate interaction between the multimedia component 1008 and the processing component 1002.

[148] The memory 1004 is configured to store various types of data to support operation in the electronic device 1000. Examples of these data include instructions for any application or method operating on the electronic device 1000, contact data, phonebook data, messages, pictures, videos, and the like. The memory 1004 may be implemented by any type of transitory or non-transitory storage device or a combination thereof, such as static random access memory (SRAM), electrically erasable programmable read only memory (EEPROM), erasable programmable read only memory (EPROM), programmable read only memory (PROM), read only memory (ROM), magnetic memory, flash memory, magnetic disk or optical disk.

[149] The power supply assembly 1006 provides power to various components of the electronic device 1000. The power component 1006 may include a power management system, one or more power supplies, and other components associated with generating, managing, and distributing power for the electronic device 1000.

[150] The multimedia component 1008 includes a screen providing an output interface between the electronic device 1000 and the user. In some embodiments, the screen may include a liquid crystal display (LCD) and a touch panel (TP). If the screen includes a touch panel, the screen may be implemented as a touch screen to receive an input signal from the user. The touch panel includes one or more touch sensors to sense touch, sliding and gestures on the touch panel. The touch sensor can not only sense the boundary of the touch or slide operation, but also detect the duration and pressure related to the touch or slide operation. In some embodiments, the multimedia component 1008 includes a front camera and / or a rear camera. When the electronic device 1000 is in an operation mode, such as shooting mode or video mode, the front camera and / or rear camera can receive external multimedia data. Each front camera and rear camera can be a fixed optical lens system or have focal length and optical zoom capability.

[151] The audio component 1010 is configured to output and/or input audio signals. For example, the audio component 1010 includes a microphone (MIC) configured to receive an external audio signal when the electronic device 1000 is in an operation mode, such as a call mode, a recording mode, and a speech recognition mode. The received audio signal may be further stored in the memory 1004 or transmitted via the communication component 1016. In some embodiments, the audio component 1010 also includes a speaker for outputting an audio signal.

[152] The I/O interface 1012 provides an interface between the processing component 1002 and the peripheral interface module, which can be a keyboard, click wheel, button, etc. These buttons may include but are not limited to: home button, volume button, start button and lock button.

[153] The sensor assembly 1014 includes one or more sensors for providing state evaluation on various aspects for the electronic device 1000. For example, the sensor assembly 1014 may detect the on/off state of the electronic device 1000 and the relative positioning of the components. For example, the components are the display and keypad of the electronic device 1000, and the sensor component 1014 can also detect the position change of the electronic device 1000 or one component of the electronic device 1000, the presence or absence of user contact with the electronic device 1000, the orientation or acceleration/deceleration of the electronic device 1000 and the temperature change of the electronic device 1000. The sensor assembly 1014 may include a proximity sensor configured to detect the presence of nearby objects without any physical contact. The sensor assembly 1014 may also include a light sensor, such as a complementary metal oxide semiconductor (CMOS) or charge-coupled device (CCD) image sensor, for use in imaging applications. In some embodiments, the sensor assembly 1014 may also include an acceleration sensor, a gyroscope sensor, a magnetic sensor, a pressure sensor, or a temperature sensor.

[154] The communication component 1016 is configured to facilitate wired or wireless communication between the electronic device 1000 and other devices. The electronic device 1000 may access a wireless network based on a communication standard, such as wireless fidelity (WiFi), the 2nd generation (2G) or the 3rd generation (3G), or a combination thereof. In an exemplary embodiment, the communication component 1016 receives a broadcast signal or broadcast related information from an external broadcast management system via a broadcast channel. In an exemplary embodiment, the communication component 1016 also includes a near field communication (NFC) module to facilitate short-range communication. For example, the NFC module can be implemented based on radio frequency identification (RFID) technology, infrared data association (IrDA) technology, ultra wideband (UWB) technology, Bluetooth (BT) technology and other technologies.

[155] In an exemplary embodiment, the electronic device 1000 may be implemented by one or more application specific integrated circuits (ASIC), digital signal processors (DSPS), digital signal processing devices (DSPD), programmable logic devices (PLD), field programmable gate arrays (FPGA), controllers, microcontrollers, microprocessors, or other electronic components for performing the above methods.

[156] In an exemplary embodiment, a non-transitory computer readable storage medium is provided, for example, a memory 1004 storing computer program instructions. The computer program instructions are operable with a processor 1020 of the electronic device 1000 to perform the above methods.

[157] FIG. 11 is a block diagram illustrating another electronic device according to embodiments of the disclosure. For example, the electronic device 1100 can be provided as a server. Referring to FIG. 11, the electronic device 1100 includes a processing component 1122, which further includes one or more processors as well as memory resources represented by a memory 1132 and configured to store instructions (such as applications) executable with the processing component 1122. The application stored in the memory 1132 may include one or more modules, each corresponding to a set of instructions. The applications stored in the processing component 1122 is configured to carry out instructions to perform the above methods.

[158] The electronic device 1100 may also include a power supply component 1126 configured to perform power management of the electronic device 1100, a wired or wireless network interface 1150 configured to connect the electronic device 1100 to the network, and an input/output (I/O) interface 1158. The electronic device 1100 may operate an operating system stored in the memory 1132, such as Windows Server™, Mac OS X™, Unix™, Linux™, FreeBSD™, etc.

[159] In an exemplary embodiment, a non-transitory computer readable storage medium is provided, such as a memory 1132 storing computer program instructions. The computer program instructions can be carried out by the processing component 1122 of the electronic device 1100 to perform the above methods.

[160] This disclosure can be a system, a method and/or a computer program product. The computer program product includes a computer readable storage medium, which carries computer readable program instructions for a processor to implement various aspects of the disclosure.

[161] The computer-readable storage medium may be a tangible device that can hold and store instructions used by the instruction execution device. The computer-readable storage medium may be, for example, but is not limited to, an electrical storage device, a magnetic storage device, an optical storage device, an electromagnetic storage device, a semiconductor storage device, or any suitable combination of the above. More specific examples (non-exhaustive list) of a computer-readable storage medium include: portable computer disk, hard disk, random access memory (RAM), read only memory (ROM),

erasable programmable read only memory (EPROM or flash memory), static random access memory (SRAM), portable compact disk read only memory (CD-ROM), digital multifunction disk (DVD), A memory stick, a floppy disk, a mechanical encoding device, such as a punch card or a convex structure in a groove on which instructions are stored, and any suitable combination of the above. The computer-readable storage medium used herein is not interpreted as an instantaneous signal itself, such as a radio wave or other freely propagating electromagnetic wave, an electromagnetic wave propagating through a waveguide or other transmission medium (for example, an optical pulse through an optical fiber cable), or an electrical signal transmitted through a wire.

[162] The computer-readable program instructions described herein may be downloaded from a computer-readable storage medium to various computing/processing devices, or to an external computer or external storage device through a network, such as the Internet, a local area network, a wide area network, and/or a wireless network. The network may include copper transmission cables, optical fiber transmission, wireless transmission, routers, firewalls, switches, gateway computers, and/or edge servers. The network adapter card or network interface in each computing/processing device receives computer-readable program instructions from the network and forwards the computer-readable program instructions to be stored in the computer-readable storage medium in each computing/processing device.

[163] The computer program instructions for performing the operations of the present disclosure may be assembly instructions, instruction set architecture (ISA) instructions, machine instructions, machine related instructions, microcode, firmware instructions, status setting data, or source code or object code written in any combination of one or more programming languages. The programming languages include object-oriented programming languages such as Smalltalk, C + +, etc., and conventional procedural programming languages such as “C” language or similar programming languages. Computer readable program instructions may be executed completely on the user’s computer, partially on the user’s computer, as a separate software package, partially on the user’s computer and partially on the remote computer, or completely on the remote computer or server. In the case of a remote computer, the remote computer may be connected to the user computer through any kind of network, including a local area network (LAN) or a wide area network (WAN), or may be connected to an external computer (for example., through an Internet service provider). In some embodiments, an electronic circuit such as a programmable logic circuit, a field programmable gate array (FPGA), or a programmable logic array (PLA) can be personalized by utilizing the state information of computer-readable program instructions, the

electronic circuit can carry out computer readable program instructions, so as to implement various aspects of the present disclosure.

[164] Various aspects of the present disclosure are described herein with reference to flow charts and/or block diagrams of methods, devices (systems) and computer program products according to embodiments of the present disclosure. It should be understood that each block of a flowchart and/or block diagram and a combination of blocks in the flowchart and/or block diagram may be implemented by computer-readable program instructions.

[165] These computer-readable program instructions can be provided to the processor of a general-purpose computer, a special-purpose computer or other programmable data processing devices, so as to produce a machine so that when these instructions are executed by the processor of a computer or other programmable data processing device, a device is generated to realize the functions/actions specified in one or more blocks in the flowchart and/or block diagram. These computer-readable program instructions may also be stored in a computer-readable storage medium that causes computers, programmable data processing devices and/or other devices to operate in a specific manner, so that the computer-readable medium in which the instructions are stored includes a manufacture, which includes instructions to implement various aspects of the functions/actions specified in one or more blocks in the flowchart and/or block diagram.

[166] Computer readable program instructions may also be loaded onto a computer, other programmable data processing devices, or other devices, so that a series of operation steps are performed on the computer, other programmable data processing device, or other devices to generate a computer implemented process, so that instructions executed in the computer, other programmable data processing device or other devices can realize the functions/actions specified in one or more blocks in the flowchart and/or block diagram.

[167] The flowchart and block diagram in the accompanying drawings show the architecture, functions and operations of possible implementations of systems, methods and computer program products according to various embodiments of the present disclosure. In this regard, each block in the flowchart or block diagram may represent a part of a module, program segment or instruction containing one or more executable instructions for realizing a specified logical function. In some alternative implementations, the functions marked in the blocks may also occur in a different order than those marked in the drawings. For example, two consecutive blocks can actually be executed basically in parallel, and they can sometimes be executed in the opposite order, depending on the function involved. It should also be noted that each block in the block diagram and/or flowchart and the combination of blocks in the

block diagram and/or flowchart can be realized by a dedicated hardware based system performing specified functions or actions, or by a combination of dedicated hardware and computer instructions.

[168] Embodiments of the present disclosure have been described above, and the above description is exemplary rather than exhaustive, and is not limited to the disclosed embodiments. Without departing from the scope and spirit of the described embodiments, many modifications and changes are obvious to those skilled in the art. The terms used herein are selected to best explain the principle, practical application or improvement of the technology in the market of each embodiment, or to enable other ordinary skilled person in the art to understand the embodiments disclosed herein.

What is claimed is:

1. A method for determining a deep learning network, comprising:
 - determining a target event to be processed by a target device;
 - determining an original network from a trained network set according to the target event, the original network being a network in the network set which matches the target event;
 - obtaining device resource information of an original device that trains the original network and device resource information of the target device; and
 - dividing a target network from the original network according to the device resource information of the target device and the device resource information of the original device, the target network being to be deployed on the target device to process the target event, the target network and the original network being the same in the number of network modules and a connection structure between network modules.
2. The method of claim 1, further comprising training to obtain the original network:
 - obtaining a sample data set, the sample data set comprising a plurality of input data and labels each corresponding to one piece of input data;
 - constructing a preset learning network, determining a plurality of sampling networks from the preset learning network, and setting an initial parameter for the sampling network; and
 - training the plurality of sampling networks with the sample data set and assigning the plurality of sampling networks trained as a plurality of standby networks, and determining the original network from the plurality of standby networks.
3. The method of claim 2, wherein the plurality of sampling networks comprises a first level network, at least one second level network, and a third level network, the first level network, the at least one second level network, and the third level network are same in the number of network modules and a connection structure between network modules;
 - each network module in the first level network having an output resolution greater than a corresponding network module in the second level network, and each network module in the second level network having an output resolution greater than a corresponding network module in the third level network;
 - each network module in the first level network having an output channel quantity

greater than a corresponding network module in the second level network, and each network module in the second level network having an output channel quantity greater than a corresponding network module in the third level network;

each network module in the first level network having a convolution module quantity greater than a corresponding network module in the second level network, and each network module in the second level network having a convolution module quantity greater than a corresponding network module in the third level network.

4. The method of claim 2 or 3, wherein training the plurality sampling networks with the sample data set and assigning the plurality sampling networks trained as the plurality of standby networks comprises:

assigning each of the plurality of sampling networks as a present sampling network;
processing the plurality of input data based on the present sampling network to determine an output data corresponding to each of the plurality of input data;

determining a loss value based on the output data and a label corresponding to the input data; and

determining the present sampling network as a standby network when the loss value is less than or equal to the preset threshold.

5. The method of claim 4, further comprising:

before determining the present sampling network as the standby network when the loss value is less than or equal to the preset threshold,

back propagating based on the loss value when the loss value is greater than the preset threshold, performing parameter update on the present sampling network to obtain an updated sampling network, and determining the updated sampling network as the present sampling network; and

repeating the following:

processing the plurality of input data based on the present sampling network to determine the output data corresponding to each of the plurality of input data; and

determining the loss value based on the output data and the label corresponding to the input data.

6. The method of claim 3, wherein training the plurality sampling networks with the

sample data set and assigning the plurality sampling networks trained as the plurality of standby networks comprises:

assigning the first level network as a present first sampling network, and assigning each of at least one second level work and the third level network as a present second sampling network;

processing the plurality of input data based on the present first sampling network to determine a first output data corresponding to each input data, and determining a first loss value based on the first output data corresponding to each input data and a label corresponding to each input data;

for each of a plurality of present second sampling networks: processing the plurality of input data based on the present second sampling network to determine a second output data corresponding to each input data, and determining a second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data;

determining the present first sampling network and the present second sampling deep learning network as the standby network, when the first loss value is less than or equal to the first preset threshold and the second loss value is less than or equal to the second preset threshold.

7. The method of claim 6, further comprising:

before determining the present first sampling network and the present second sampling deep learning network as the standby network when the first loss value is less than or equal to the first preset threshold and the second loss value is less than or equal to the second preset threshold,

back propagating based on the first loss value when the first loss value is greater than the first preset threshold, performing parameter update on the present first sampling network to obtain an updated first sampling network, and determining the updated first sampling deep learning network as the present first sampling network;

back propagating based on the second loss value when the second loss value is greater than the second preset threshold, performing parameter update on the present second sampling network to obtain an updated second sampling network, and determining the updated second sampling deep learning network as the present second sampling network; and

repeating the following:

processing the plurality of input data based on the present first sampling network to determine the first output data corresponding to each input data, and determining the first loss value based on the first output data corresponding to each input data and the label corresponding to each input data; and

processing the plurality of input data based on the present second sampling network to determine the second output data corresponding to each input data, and determining the second loss value based on the second output data corresponding to each input data and the first output data corresponding to each input data.

8. The method of claim 1 or 2, wherein the original network comprises one main branch and at least one slave branch, the main branch comprises a plurality of network modules, and network modules in each slave branch comprises part of the network modules in the main branch, wherein

if the original network has a plurality of slave branches, each of the plurality of branches shares at least one network module.

9. The method of claim 8, wherein each network module comprises at least one convolutional module, and each convolutional module comprises one convolutional layer, one batch normalization (BN) layer, and one activation layer, wherein

the convolutional layer, the BN layer, and the activation layer are connected in series;

the convolutional layer is a depthwise convolutional layer, a pointwise convolutional layer, or a two-dimensional convolutional layer.

10. The method of claim 1, wherein the device resource information of the original device at least comprises: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the original device, the device resource information of the target device at least comprises: video-memory capacity data, video-memory performance data, network performance data, memory capacity data, and computing ability data of the target device.

11. The method of claim 1, wherein determining the original network from the trained network set according to the target event comprises:

decomposing the target event to determine a target object identifier and a target task identifier of the target event;
obtaining a task identifier and an object identifier of each network in the network set;
determining a first match degree value between the target task identifier and each task identifier;
determining a second match degree value between the target object identifier and each object identifier; and
determining the original network from the network set according to the first match degree value and the second match degree value.

12. The method of claim 1, wherein dividing the target network from the original network according to the device resource information of the target device and the device resource information of the original device comprises:

adjusting an output resolution, an output channel quantity, and a convolutional module quantity of each network module in the original network according to the device resource information of the target device and the device resource information of the original device, with a preset search rule; and
determining the original deep learning network adjusted as the original network.

13. An apparatus for determining a deep learning network, comprising:

a target event determining unit configured to determine a target event to be processed by a target device;

an original deep learning network determining unit configured to determine an original network from a trained network set according to the target event, the original network being a network in the network set which matches the target event;

a resource information obtaining unit configured to obtain device resource information of an original device that trains the original network and device resource information of the target device;

a target network dividing unit configured to divide a target network from the original network according to the device resource information of the target device and the device resource information of the original device, the target network being deployed on the target device to process the target event, the target network and the original network being the same in the number of network modules and a connection structure between network modules.

14. An electronic device, comprising at least one processor and a memory in communication with the at least one processor, the memory storing instructions which are executable with the at least one processor, the processor being configured to execute the instructions stored in the memory to perform the method of any of claims 1 to 12.

15. A computer readable storage medium, storing at least one instruction or at least one program, the at least one instruction or the at least one program being uploaded and executed by a processor to perform the method of any of claims 1 to 12.