

# LEARNING DEVICE, LEARNING METHOD, AND LEARNING PROGRAM FIELD

[0001] The present invention relates to a learning device, a learning method, and a learning program.

## BACKGROUND

[0002] A known deep learning training method using a neural network selects, from prepared images, an image used as labeled data to allow efficient training of a model and augments the image (e.g., Patent Literature 1). The known learning method including specifically designed data augmentation and data selection processes can be used for training using large-scale data.

[0003] The known learning method uses, in the data selection process, an additional condition to avoid selecting the same image as selected before the data augmentation process. This reduces the likelihood of similar images, increased through the data augmentation process, being used for duplicate training and thus prevents training images from being biased. However, the known learning method uses fewer images as labeled data per training and thus cannot allow sufficient learning of the characteristics of the images. This can increase the loops for training and taking more time to complete the training.

## CITATION LIST

### PATENT LITERATURE

[0004] Patent Literature 1: Japanese Unexamined Patent Application Publication No. 2024-002431

## BRIEF SUMMARY

### TECHNICAL PROBLEM

[0005] One or more aspects of the present invention are directed to a learning device, a learning method, and a learning program that allow fast training.

### SOLUTION TO PROBLEM

[0006] (1) A learning device according to one aspect of the present invention includes a data augmentation processor that creates pieces of training-candidate data increased based on pieces of source image data, a confidence evaluator that assigns confidence scores to the pieces

of training-candidate data, a training data extractor that extracts, from the pieces of training-candidate data, pieces of training data in order of the confidence scores, and a training processor that performs training based on the pieces of training data. The training data extractor extracts the pieces of training data up to a piece of training-candidate data assigned with a confidence score equal to a preset confidence score threshold.

(2) In (1), the learning device may include a representative image selector that selects, from the pieces of training-candidate data, one or more representative images each representing a group of augmented images having common source image data.

(3) In (2), the training data extractor may extract the pieces of training data until a number of types of the one or more representative images reaches a preset threshold for the number of types of representative images.

(4) In (2), the training data extractor may add, to the pieces of training data, pieces of training-candidate data assigned with lower confidence scores than a piece of training-candidate data assigned with a highest confidence score of the one or more representative images.

(5) A learning method according to one aspect of the present invention includes a data augmentation step of creating pieces of training-candidate data increased based on pieces of source image data, a confidence evaluation step of assigning confidence scores to the pieces of training-candidate data, a training data extraction step of extracting, from the pieces of training-candidate data, pieces of training data in order of the confidence scores, and a training step of performing training based on the pieces of training data. The training data extraction step includes extracting the pieces of training data up to a piece of training-candidate data assigned with a confidence score equal to a preset confidence score threshold.

(6) A learning program according to one aspect of the present invention causes a computer to execute a data augmentation function of creating pieces of training-candidate data increased based on pieces of source image data, a confidence evaluation function of assigning confidence scores to the pieces of training-candidate data, a training data extraction function of extracting, from the pieces of training-candidate data, pieces of training data in order of the confidence scores, and a training function of performing training based on the pieces of training data. The training data extraction function includes extracting the pieces of training data up to a piece of

training-candidate data assigned with a confidence score equal to a preset confidence score threshold.

### ADVANTAGEOUS EFFECTS

[0007] The learning device, the learning method, and the learning program according to the above aspects of the present invention allow fast training.

### BRIEF DESCRIPTION OF DRAWINGS

[0008] FIG. 1 is a block diagram of a learning device.

FIG. 2 is a flowchart of a learning method.

FIG. 3 is a conceptual diagram of the learning method including extracting a limited number of pieces of training data based on the number of representative images.

FIG. 4 is a conceptual diagram of the learning method including extracting a limited number of pieces of training data based on a confidence score threshold.

### DETAILED DESCRIPTION

#### Embodiments

[0009] One or more embodiments of the present invention will be described in detail below with reference to the drawings.

FIG. 1 is a block diagram of a learning device 100. FIG. 2 is a flowchart of a learning method. FIG. 3 is a conceptual diagram of the learning method including extracting a limited number of pieces of training data based on the number of representative images. FIG. 4 is a conceptual diagram of the learning method including extracting a limited number of pieces of training data based on a confidence score threshold  $\alpha$ . Like reference numerals or signs may denote the components having a common capability.

#### Learning device

[0010] The learning device 100 according to the present embodiment will now be described with reference to FIG. 1. As shown in FIG. 1, the learning device 100 according to the present embodiment includes a data augmentation processor 10, a confidence evaluator 20, a representative image selector 30, a training data extractor 40, a training processor 50, and a controller 60.

[0011] The learning device 100 trains a machine learning model (hereafter referred to as a

model) to evaluate target data (e.g., image data) accurately. The model herein refers to a mechanism (algorithm) in machine learning that receives an input value identifiable by a computer and outputs a score.

[0012] The model identifies and evaluates the details of the target data. For the target data that is image data, for example, the model evaluates the quality of the subject (e.g., as an acceptable product or a rejected product).

[0013] The model includes a neural network and parameters. The neural network includes multiple neurons connected to one another. The neural network may be, for example, a hierarchical multilayer neural network with a series of layers in each of which multiple neurons are grouped. The neural network is defined by the number of neurons and the connection relationship between the neurons. The strength of connection between the neurons or the layers is defined using parameters (e.g., weight coefficients). In the neural network, input target data is evaluated based on the results of calculation performed by the multiple neurons and the parameters. For example, the model classifies pieces of target data (training data) as data assigned with an acceptable-product label or data assigned with a rejected-product label, or calculates acceptable-product scores for pieces of target data (training data).

[0014] The learning device 100 includes a computer (not shown) as its main hardware. The computer includes a processor such as a central processing unit (CPU) or a graphics processing unit (GPU), a storage such as a random-access memory (RAM), a read-only memory (ROM), a hard disk drive (HDD), or a solid-state drive (SSD), and a communicator that allows connection to a network such as the Internet. The storage in the computer stores a learning program for causing the computer to operate as the learning device 100 according to the present embodiment. The computer includes a cloud computing system. The functional components of the learning device 100 are implemented by the computer as hardware and the learning program as software.

[0015] The data augmentation processor 10 is a functional component that generates pieces of training-candidate data (a training-candidate dataset) increased based on pieces of source image data. The process of increasing data can increase the quality and quantity of image data to augment the image data, and can generate, from a small amount of data, a sufficiently large

amount of data with high quality and efficiency for training a machine learning model, thus improving the performance of the model. Each piece of source image data is increased by, for example, flipping, shifting, slight rotation, or filtering. Flipping creates data by flipping the source image data horizontally or vertically or rotating the source image data by 180°.

5 Shifting creates data by shifting a coordinate of the source image data in the X-direction or the Y-direction. Slight rotation creates data by rotating the source image data by a small angle in the right direction or the left direction. Filtering creates data by applying, for example, a median filter or a Gaussian filter to the source image data.

[0016] The confidence evaluator 20 is a functional component that assigns, to each piece  
10 of training-candidate data, a confidence score  $E_s$  calculated with a predetermined formula. The confidence score  $E_s$  is an index indicating the degree by which training with the data expectedly improves the determination performance of the model. In other words, the confidence score  $E_s$  is an index indicating the degree of confidence of the model in evaluating (performing determination of) the data at the time the model is trained with the data.

15 Confidence Score

[0017] To determine the confidence score  $E_s$ , the learning device 100 (confidence evaluator 20) calculates, for each piece of training-candidate data, an acceptable-product distance based on the corresponding features. The acceptable-product distance is a distance in a feature space between the training-candidate data and acceptable-product data that is training data assigned  
20 with an acceptable-product label (labeled data used in machine learning of the model). As described above, the confidence score  $E_s$  may be a parameter calculated based on the distance between the training-candidate data and the training data in the feature space. Hereafter, each of the acceptable-product distance and a rejected-product distance may be, for example, the Euclidean distance between pieces of data projected in the feature space. The distances are  
25 not limited to the Euclidean distances, and may be any distances, such as the Mahalanobis distances, that can be calculated in the feature space. The distance between one piece of training data in a training data group and one piece of training-candidate data in a training-candidate data group is calculated with, for example, Formula 1 below.

$$d_{(k,s)}(\vec{p}_s, \vec{q}_k) = \sqrt{\sum_{i=1}^j (q_{(k,i)} - p_{(s,i)})^2}$$

(1)

In this formula,  $q(k,i)$  is the coordinate of training data  $k$  in a specific dimension  $i$  of the feature space,  $p(s,i)$  is the coordinate of the training-candidate data in the specific dimension  $i$  of the feature space, and  $d(k,s)$  is the distance between the training data  $k$  and the training-candidate data. The vector of  $q_k$  is a set of coordinate data of the training data  $k$  in the feature space, and the vector of  $p_k$  is a set of coordinate data of the training-candidate data in the feature space. Also,  $k$  is an integer less than or equal to the number of pieces of training data ( $m + n$ , where  $m$  and  $n$  are integers),  $i$  is an integer less than or equal to the number ( $j$ ) of predetermined dimensions ( $j$  is an integer), and  $s$  is an integer less than or equal to the number ( $t$ ) of pieces of training-candidate data ( $t$  is an integer).

[0018] The distance between the training-candidate data and acceptable-product data  $OK_g$  as one piece of acceptable-product data  $OK$ , or  $d(OK_g,s)$ , is expressed by Formula 2 below using Formula 1. For  $OK_g$ ,  $OK$  indicates an acceptable product, and  $g$  is an integer less than or equal to the number ( $m$ ) of pieces of acceptable-product data  $OK$ .

15

$$d_{(OK_g,s)}(\vec{p}_s, \vec{q}_{OK_g}) = \sqrt{\sum_{i=1}^j (q_{(OK_g,i)} - p_{(s,i)})^2} \quad (2)$$

In this formula,  $q(OK_g,i)$  is the coordinate of the acceptable-product data  $OK_g$  in the training data group in the specific dimension  $i$  of the feature space, and the vector of  $q_{OK_g}$  is a set of coordinate data of the acceptable-product data  $OK_g$  in the feature space.

20 [0019] The vector of  $d(OK,s)$ , which is a set of distances between the training-candidate data and the pieces of acceptable-product data  $OK$ , is expressed by Formula 3 below using Formula 2.

$$\overrightarrow{d_{(OK,s)}} = (d_{(OK1,s)}, d_{(OK2,s)}, \dots, d_{(OKm,s)}) \quad (3)$$

25 [0020] The training-candidate data has an acceptable-product distance  $E(OK,s)$  that is, for example, the minimum value of the elements of the vector of  $d(OK,s)$ . More specifically, the acceptable-product distance  $E(OK,s)$  is the minimum value of the elements of the vector of  $d(OK,s)$  that is the set of distances between the training-candidate data and the pieces of acceptable-product data  $OK$ . The acceptable-product distance  $E(OK,s)$  is expressed by

Formula 4 below using Formula 3. A smaller acceptable-product distance  $E(OK,s)$  indicates the training-candidate data closer to one of the pieces of acceptable-product data OK in the feature space.

$$E_{(OK,s)} = \min \left\{ f(x) \mid x \in \overline{d_{(OK,s)}} \right\} \quad (4)$$

[0021] The acceptable-product distance  $E(OK,s)$  of the training-candidate data may be, for example, an average of the a smallest elements extracted from the elements of the vector of  $d(OK,s)$ , where a is a natural number and is, for example, 3. In this case, the acceptable-product distance  $E(OK,s)$  is expressed by Formula 5 below using Formula 3. A smaller acceptable-product distance  $E(OK,s)$  indicates the training-candidate data closer to multiple (a) pieces of acceptable-product data OK in the feature space and the training-candidate data closer to a cluster of the pieces of acceptable-product data OK (an acceptable-product cluster).

$$E_{(OK,s)} = \frac{1}{a} \sum_{l=1}^a \min \left\{ f(x) \mid x \in \overline{d_{(OK,s)}} \right\} \quad (5)$$

[0022] The learning device 100 (confidence evaluator 20) calculates, for each piece of training-candidate data, a rejected-product distance based on the corresponding features. The rejected-product distance is a distance in the feature space between the training-candidate data and rejected product data that is training data assigned with a rejected-product label. The distance between the training-candidate data and rejected product data NGh among pieces of rejected product data NG, or  $d(NGh,s)$ , is expressed by Formula 6 below using Formula 1. For NGh, NG indicates a rejected product, and h is an integer less than or equal to the number (n) of pieces of rejected product data NG.

$$d_{(NGh,s)}(\vec{p}_s, \vec{q}_{NGh}) = \sqrt{\sum_{i=1}^j (q_{(NGh,i)} - p_{(s,i)})^2} \quad (6)$$

In this formula,  $q(NGh,i)$  is the coordinate of the rejected product data NGh, among the training data, in the specific dimension i of the feature space, and the vector of  $q_{NGh}$  is a set of coordinate data of the rejected product data NGh in the feature space.

[0023] The vector of  $d(NG,s)$ , which is a set of distances between the training-candidate data and the pieces of rejected product data NG, is expressed by Formula 7 below using Formula

6.

$$\overrightarrow{d_{(NG,s)}} = (d_{(NG1,s)}, d_{(NG2,s)}, \dots, d_{(NGm,s)}) \quad (7)$$

[0024] The training-candidate data has a rejected-product distance  $E(NG,s)$  that is, for example, the minimum value of the elements of the vector of  $\overrightarrow{d(NG,s)}$ . More specifically, the rejected-product distance  $E(NG,s)$  is the minimum value of the distances between the training-candidate data and the pieces of rejected product data NG. The rejected-product distance  $E(NG,s)$  is expressed by Formula 8 below using Formula 7. A smaller rejected-product distance  $E(NG,s)$  indicates the training-candidate data closer to one of the pieces of rejected product data NG in the feature space.

$$E_{(NG,s)} = \min \{ f(x) \mid x \in \overrightarrow{d_{(NG,s)}} \} \quad (8)$$

[0025] The rejected-product distance  $E(NG,s)$  of the training-candidate data may be, for example, an average of the a smallest elements extracted from the elements of the vector of  $\overrightarrow{d(NG,s)}$ . In this case, the rejected-product distance  $E(NG,s)$  is expressed by Formula 9 below using Formula 7. A smaller rejected-product distance  $E(NG,s)$  indicates the training-candidate data closer to multiple (a) pieces of rejected product data NG in the feature space and the training-candidate data closer to a cluster of the pieces of rejected product data NG (a rejected-product cluster).

$$E_{(NG,s)} = \frac{1}{a} \sum_{l=1}^a \min_l \{ f(x) \mid x \in \overrightarrow{d_{(NG,s)}} \} \quad (9)$$

[0026] The learning device 100 (confidence evaluator 20) calculates the confidence score  $E_s$  for the training-candidate data using the calculated acceptable-product distance  $E(OK,s)$  and rejected-product distance  $E(NG,s)$ . The confidence score  $E_s$  is, for example, a value obtained by dividing the acceptable-product distance  $E(OK,s)$  by the rejected-product distance  $E(NG,s)$ , and is expressed by Formula 10 below.

$$E_s = \frac{E_{(OK,s)}}{E_{(NG,s)}} \quad (10)$$

[0027] For example, a lower confidence score  $E_s$  less than 1 indicates that the acceptable-

product distance  $E(OK,s)$  is smaller than the rejected-product distance  $E(NG,s)$  and that the training-candidate data is closer to the acceptable-product cluster than the rejected-product cluster. For training-candidate data with a rejected-product label, a lower confidence score  $E_s$  indicates that the training-candidate data is more difficult to be classified by the model as either data to be assigned with an acceptable-product label or data to be assigned with a rejected-product label based on the training results obtained using the current training data group. This indicates that the training with such training-candidate data is more effective for the model.

[0028] In contrast, for example, a higher confidence score  $E_s$  greater than 1 indicates that the rejected-product distance  $E(NG,s)$  is smaller than the acceptable-product distance  $E(OK,s)$  and that the training-candidate data is closer to the rejected-product cluster than the acceptable-product cluster. For training-candidate data with an acceptable-product label, a higher confidence score  $E_s$  indicates that the training-candidate data is more difficult to be classified by the model as either data to be assigned with an acceptable-product label or data to be assigned with a rejected-product label based on the training results obtained using the current training data group. This indicates that the training with such training-candidate data is more effective for the model.

[0029] The representative image selector 30 is a functional component that selects, from the pieces of training-candidate data, one or more representative images each representing a group of augmented images having common source image data. As shown in FIGs. 3 and 4, for example, a single group of augmented images includes, among the pieces of training-candidate data, three pieces of data showing a circular mark. Of the three, a data piece with the minimum confidence score  $E_s$  (in ascending order) is selected as a representative image of this group of augmented images. The representative image selector 30 selects, from the pieces of training-candidate data arranged in ascending order of the confidence score  $E_s$ , the first piece of data showing a circle, the third piece of data showing a triangle, the sixth piece of data showing a rhombus, and up to the  $n$ -th data showing a chestnut burr.

[0030] The training data extractor 40 is a functional component that calculates the confidence scores  $E_s$  for the training-candidate dataset as a population and performs an extraction process to extract, as training data, pieces of training-candidate data with confidence

scores  $E_s$  within a predetermined range (e.g., less than or equal to a predetermined confidence score threshold  $\alpha$ ). In other words, the training data extractor 40 extracts, from the pieces of training-candidate data, pieces of training data in order of the confidence scores  $E_s$ .

[0031] The training processor 50 is a functional component that causes the model to undergo a training process at a predetermined learning rate. The training process is performed to adjust the parameters of the model to be closer to their optimal values.

[0032] The controller 60 is a functional component that controls the training data extractor 40 and the training processor 50 to train the model effectively.

#### Learning method

[0033] The procedure implementing the learning method according to one or more embodiments will now be described. The learning method is implementable using the learning device 100. The steps (processes or operations) in the learning method can be performed by the components (modules) of the learning device 100 corresponding to the steps. FIG. 2 is a flowchart of the learning method. FIG. 3 is a diagram showing the learning method including extracting a limited number of pieces of training data based on the number of representative images. FIG. 4 is a diagram showing the learning method including extracting a limited number of pieces of training data based on the confidence score threshold  $\alpha$ .

[0034] (1) The learning device 100 obtains a dataset used as pieces of source image data in advance.

[0035] (2) As shown in FIG. 2, the learning device 100 creates, with the data augmentation processor 10, pieces of training-candidate data (a training-candidate dataset) increased based on the pieces of source image data (data augmentation step) (process A). This process can increase the quality and quantity of image data to augment the image data, and can generate, from a small amount of data, a sufficiently large amount of data with high quality and efficiency for training a machine learning model, thus improving the performance of the model.

[0036] (3) The learning device 100 then assigns, with the confidence evaluator 20, confidence scores  $E_s$  calculated with a predetermined formula to the pieces of training-candidate data (confidence evaluation step). The learning device 100 selects, with the representative image selector 30, one or more representative images each representing a group

of augmented images having common source image data from the training-candidate dataset, using the confidence scores  $E_s$  assigned to the pieces of training-candidate data (images) (process B). Each representative image may be an image assigned with the minimum confidence score  $E_s$  among multiple pieces of image data augmented from common source image data (a group of augmented images) (the other images having the same source of augmentation are not selected in process B). In other words, an image having the minimum score is selected as a representative image in process B for images arranged in ascending order of the score, and other subsequent images having the same source image data are not selected as representative images. For example, the image showing a circle in the first row of the column of the training-candidate data is selected as a representative image, and the images showing a circle having the same source image data in the second and fourth rows are not selected as representative images, as shown in FIG. 3.

[0037] (3-1) The training data extractor 40 may extract pieces of training data up to a piece of training-candidate data assigned with a confidence score  $E_s$  equal to the preset confidence score threshold  $\alpha$  (training data extraction step).

[0038] (4) The number of representative images selected in process B is then detected. When one or more representative images are detected, the processing advances to process D. When no representative image is detected, a process to end the training is performed (process C).

[0039] (5) The selected representative images are sorted in order of the confidence scores  $E_s$  (e.g., in ascending order), and one or more representative images with lower confidence scores  $E_s$  than a predetermined score are classified as training data and one or more representative images with higher confidence scores  $E_s$  than the predetermined score are classified as data other than the training data (process D).

[0040] (5-1) In process D, the learning device 100 may select one or more representative images up to a representative image with a confidence score  $E_s$  equal to the preset confidence score threshold  $\alpha$  (condition 1). This reduces the likelihood of pieces of inefficient training-candidate data for training being extracted as training data (labeled data), and also allows extraction of an appropriate amount of training data that is not too small and not too large.

[0041] (5-2) In process D, the learning device 100 may perform the classification process until the number of representative images reaches a predefined number (condition 2). In other words, the training data extractor 40 in the learning device 100 extracts pieces of training data until the number of types of representative images reaches a preset threshold  $\beta$  for the number of types of representative images. This allows extraction of an appropriate amount of training data that is not too small and not too large.

[0042] (5-3) In process D, the learning device 100 may classify all the representative images (condition 3). This allows extraction of an appropriate amount of training data that is not too small and not too large.

The conditions 1 to 3 described above may be applied independently or in combination as appropriate.

[0043] (6) The representative images classified as training data in process D are extracted as pieces of training data (process E).

[0044] (6-1) Images with lower confidence scores  $E_s$  than the representative images are extracted as pieces of training data (process G). In other words, the training data extractor 40 adds, to the training data, pieces of training-candidate data assigned with lower confidence scores  $E_s$  than the piece of training-candidate data assigned with the highest confidence score  $E_s$  of the representative images. This allows extraction of all the images in each group of augmented images having source image data in common with their representative images.

This reduces insufficient extraction of training data and allows extraction of a sufficient amount of training data more effective for training, compared with when the other pieces of training-candidate data having source image data in common with their representative images are not extracted.

[0045] (7) After process E, training is performed based on the pieces of training data extracted in the processes up to process E (process F). A condition is set to end the training when the training is performed more than a predetermined number of times for appropriate training. The processing returns to process B after the training is performed in process F.

[0046] (8) A pre-trained model is set as an initial value between processes A and B as appropriate (process H).

[0047] As described above, the learning method according to the present embodiment extracts, using the learning device 100, pieces of training data based on representative images for pieces of source image data used as the sources of data augmentation until one or more predetermined conditions (the above conditions 1 to 3) are satisfied. This allows reliable  
5 extraction of pieces of quality training data of different types for training, and also allows extraction of a sufficient, but not too large amount of training data for training. The learning device 100 and the learning method thus allow fast training.

#### Learning program

[0048] The learning program causes the computer to perform the processing performed in  
10 each step in the above learning method. More specifically, the learning program causes a data augmentation function, a confidence evaluation function, a training data extraction function, and a training function to correspond to the steps in the above learning method. This generates, from a small amount of data, a sufficiently large amount of data with high quality and efficiency for training a machine learning model, thus improving the performance of the model. The  
15 learning method allows fast training.

[0049] The technical scope of the present invention is not limited to the above embodiments, and various changes may be made without departing from the spirit and scope of the present invention.

[0050] The components in the above embodiments may be replaced with known  
20 components as appropriate without departing from the spirit and scope of the present invention. The modifications described above may be combined as appropriate without departing from the spirit and scope of the present invention.

[0051] As described above, the learning device 100 according to one or more embodiments includes the data augmentation processor 10 that creates pieces of training-candidate data  
25 increased based on pieces of source image data, the confidence evaluator 20 that assigns the confidence scores  $E_s$  to the pieces of training-candidate data, the training data extractor 40 that extracts, from the pieces of training-candidate data, pieces of training data in order of the confidence scores  $E_s$ , and the training processor 50 that performs training based on the pieces of training data. The training data extractor 40 extracts the pieces of training data up to a piece

of training-candidate data assigned with a confidence score  $E_s$  equal to the preset confidence score threshold  $\alpha$ . This generates, from a small amount of data, a sufficiently large amount of data with high quality and efficiency for training a machine learning model, thus improving the performance of the model. The learning device 100 allows fast training.

5 [0052] The learning method according to one or more embodiments includes a data augmentation step of creating pieces of training-candidate data increased based on pieces of source image data, a confidence evaluation step of assigning the confidence scores  $E_s$  to the pieces of training-candidate data, a training data extraction step of extracting, from the pieces of training-candidate data, pieces of training data in order of the confidence scores  $E_s$ , and a  
10 training step of performing training based on the pieces of training data. In the learning method, the training data extraction step includes extracting the pieces of training data up to a piece of training-candidate data assigned with a confidence score  $E_s$  equal to the preset confidence score threshold  $\alpha$ . This generates, from a small amount of data, a sufficiently large amount of data with high quality and efficiency for training a machine learning model, thus  
15 improving the performance of the model. The learning method allows fast training.

[0053] The learning program according to one or more embodiments causes the computer to execute a data augmentation function of creating pieces of training-candidate data increased based on pieces of source image data, a confidence evaluation function of assigning the confidence scores  $E_s$  to the pieces of training-candidate data, a training data extraction function  
20 of extracting, from the pieces of training-candidate data, pieces of training data in order of the confidence scores  $E_s$ , and a training function of performing training based on the pieces of training data. In the learning program, the training data extraction function includes extracting the pieces of training data up to a piece of training-candidate data assigned with a confidence score  $E_s$  equal to the preset confidence score threshold  $\alpha$ . This generates, from a small amount  
25 of data, a sufficiently large amount of data with high quality and efficiency for training a machine learning model, thus improving the performance of the model. The learning program allows fast training.

## REFERENCE SIGNS LIST

[0054]

100 learning device

10 data augmentation processor

5 20 confidence evaluator

30 representative image selector

40 training data extractor

50 training processor

60 controller

10  $\alpha$  confidence score threshold

$\beta$  threshold for the number of types of representative images

## CLAIMS

1. A learning device, comprising:

a data augmentation processor configured to create pieces of training-candidate data increased based on pieces of source image data;

5 a confidence evaluator configured to assign confidence scores to the pieces of training-candidate data;

a training data extractor configured to extract, from the pieces of training-candidate data, pieces of training data in order of the confidence scores;

a training processor configured to perform training based on the pieces of training data;

10 and

a representative image selector configured to select, from the pieces of training-candidate data, one or more representative images each representing a group of augmented images having common source image data, and extract the one or more representative images as the pieces of training data,

15 wherein the training data extractor extracts the pieces of training data up to a piece of training-candidate data assigned with a confidence score equal to a preset confidence score threshold.

2. The learning device according to claim 1, wherein

the training data extractor extracts the pieces of training data until a number of types of the

one or more representative images reaches a preset threshold for the number of types of representative images.

3. The learning device according to claim 1, wherein

the training data extractor adds, to the pieces of training data, pieces of training-candidate

5 data assigned with lower confidence scores than a piece of training-candidate data assigned with a highest confidence score of the one or more representative images.

4. A learning method, comprising:

a data augmentation step of creating pieces of training-candidate data increased based on pieces of source image data;

10 a confidence evaluation step of assigning confidence scores to the pieces of training-candidate data;

a training data extraction step of extracting, from the pieces of training-candidate data, pieces of training data in order of the confidence scores;

a training step of performing training based on the pieces of training data; and

15 a representative image selection step of selecting, from the pieces of training-candidate data, one or more representative images each representing a group of augmented images having common source image data, and extracting the one or more representative images as the pieces of training data,

wherein the training data extraction step includes extracting the pieces of training data up

to a piece of training-candidate data assigned with a confidence score equal to a preset confidence score threshold.

5. A learning program for causing a computer to execute:

a data augmentation function of creating pieces of training-candidate data increased based

5 on pieces of source image data;

a confidence evaluation function of assigning confidence scores to the pieces of training-candidate data;

a training data extraction function of extracting, from the pieces of training-candidate data, pieces of training data in order of the confidence scores;

10 a training function of performing training based on the pieces of training data; and

a representative image selection function of selecting, from the pieces of training-candidate data, one or more representative images each representing a group of augmented images having common source image data, and extracting the one or more representative images as the pieces of training data,

15 wherein the training data extraction function includes extracting the pieces of training data up to a piece of training-candidate data assigned with a confidence score equal to a preset confidence score threshold.