(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
18 January 2018 (18.01.2018)

WIPO | PCT

(10) International Publication Number
WO 2018/013982 A1

(51) International Patent Classification:
G06F 17/30 (2006.01) *G06N 3/08* (2006.01)
G06N 3/04 (2006.01) *G06K 9/66* (2006.01)
G06N 3/063 (2006.01)

(21) International Application Number:
PCT/US2017/042235

(22) International Filing Date:
14 July 2017 (14.07.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/362,488 14 July 2016 (14.07.2016) US

(71) Applicant: **GOOGLE LLC** [US/US]; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).

(72) Inventor: **CHOLLET, Francois**; 1600 Amphitheatre Parkway, Mountain View, California 94043 (US).

(74) Agent: **PORTNOV, Michael** et al.; FISH & RICHARDSON P.C., P.O. BOX 1022, 3300 RBC PLAZA, MINNEAPOLIS, Minnesota 55440-1022 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(54) Title: CLASSIFYING IMAGES USING MACHINE LEARNING MODELS

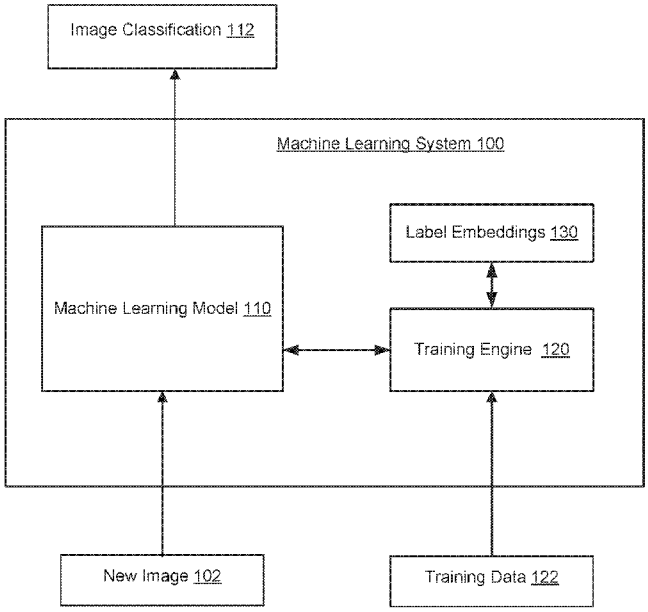

FIG. 1

(57) Abstract: Systems and methods for classifying an image using a machine learning model. One of the methods includes obtaining training data for training the machine learning model, wherein the machine learning model is configured to process input images to generate, for each input image, a predicted point in an embedding space; determining, from label data for training images in the training data, a respective numeric embedding of each of the object categories, wherein a distance in the embedding space between the numeric embeddings of any two object categories reflects a degree of visual co-occurrence of the two object categories; and training the machine learning model on the training data. The systems described in this specification can effectively perform multi-label, massively multi-category image classification, where the number of classes is large (many thousands or tens of thousands) and where each image typically belongs to multiple categories that should all be properly identified.

WO 2018/013982 A1

[Continued on next page]

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

CLASSIFYING IMAGES USING MACHINE LEARNING MODELS

BACKGROUND

[1] This specification relates to processing images using machine learning models.

[2] Image classification systems can identify objects in images, i.e., classify input images as including objects from one or more object categories. Some image classification systems use one or more machine learning models, e.g., deep neural networks, to classify an input image.

[3] Machine learning models receive an input and generate an output, e.g., a predicted output, based on the received input. Some machine learning models are parametric models and generate the output based on the received input and on values of the parameters of the model.

[4] Some machine learning models are deep models that employ multiple layers of models to generate an output for a received input. For example, a deep neural network is a deep machine learning model that includes an output layer and one or more hidden layers that each apply a non-linear transformation to a received input to generate an output.

SUMMARY

[5] This specification describes how a system implemented as computer programs on one or more computers in one or more locations can train a machine learning model and, once trained, use the trained machine learning model to classify received images.

[6] Particular embodiments of the subject matter described in this specification can be implemented so as to realize one or more of the following advantages. The image classification system described in this specification can effectively perform multi-label, massively multi-category image classification, where the number of classes is large (many thousands or tens of thousands) and where each image typically belongs to multiple categories that should all be properly identified. In particular, by generating numeric embeddings of object categories as described in this specification and classifying images using these embeddings, the image classification system is able to accurately classify input images even when the images include objects belonging to multiple object classes. In particular, by exploiting the internal structure of the category space to generate the embeddings based on category co-occurrences, gains in one or more of training speed,

precision, or recall of the machine learning model that is used by the classification system can be achieved.

[7] The details of one or more embodiments of the subject matter of this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the subject matter will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[8] FIG. 1 is a block diagram of an example of an image classification system.

[9] FIG. 2 is a flow diagram of an example process for training a machine learning model to classify images.

[10] FIG. 3 is a flow diagram of an example process for classifying a new image using a trained machine learning model.

[11] Like reference numbers and designations in the various drawings indicate like elements.

DETAIL DESCRIPTION

[12] This specification describes how a system implemented as computer programs on one or more computers in one or more locations can determine numeric embeddings of object categories in an embedding space, use the numeric embeddings to train a machine learning model to classify images, and, once trained, use the trained machine learning model to classify received images.

[13] FIG. 1 shows an example image classification system 100. The image classification system 100 is an example of a system implemented as computer programs on one or more computers in one or more locations in which the systems, components, and techniques described below are implemented.

[14] The image classification system 100 uses a machine learning model 110 and label embedding data 130 to classify received images. For example, the image classification system 100 can receive a new image 102 and classify the new image 102 to generate image classification data 112 that identifies one or more object categories from a predetermined set of object categories to which one or more objects depicted in the new image 102 belong. Once generated, the system 100 can store the image classification data 112 in association with the new image 102 in a data store, provide the image classification data 112 as input to another system for further processing, or transmit the image classification data 112 to a user

of the system, e.g., transmit the image classification data 112 over a data communication network to a user device from which the new image 102 was received.

[15] In particular, the label embedding data 130 is maintained by the image classification system 100, e.g., in one or more databases, and is data that maps each object category in the set of object categories to a respective numeric embedding of the object category in an embedding space. The embedding space is a *k*-dimensional space of numeric values, e.g., floating point values or quantized floating point values. Generally, *k* is a fixed integer value, e.g., a value on the order of one thousand or more. For example, in some cases, *k* may be equal to 4096 and each point in the embedding space is therefore a 4096-dimensional point.

[16] The machine learning model 100 is a model, e.g., a deep convolutional neural network, that is configured to process input images to generate, for each input image, a predicted point in the embedding space, i.e., a *k*-dimensional point.

[17] To classify the new image 102, the system 100 processes the new image 102 using the machine learning model 110 to generate a predicted point in the embedding space for the new image. The system 100 then determines one or more numeric embeddings that are closest to the predicted point from among the numeric embeddings in the label embedding data 102 and classifies the new image 102 as including images of one or more objects that belong to the object categories represented by the one or more closest numeric embeddings. Classifying new images is described in more detail below with reference to FIG. 3.

[18] To allow the model 110 to be used to effectively classify input images, the system 100 include a training engine 120 that receives training data 122 and uses the training data 122 to generate the numeric embeddings of the object categories and to train the machine learning model 110.

[19] Generally, the training engine 120 generates the numeric embeddings such that a distance in the embedding space between the numeric embeddings for any two object categories reflects a degree of visual co-occurrence of the two object categories in images and then uses the generated embeddings to train the machine learning model 110. Generating numeric embeddings and training a machine learning model is described in more detail below with reference to FIG. 2.

[20] FIG. 2 is a flow diagram of an example process 200 for training a machine learning model to classify images.

[21] For convenience, the process 200 will be described as being performed by a system of one or more computers located in one or more locations. For example, an image

classification system, e.g., the image classification system 100 of FIG. 1, appropriately programmed in accordance with this specification, can perform the process 200.

[22] The system receives training data for training a machine learning model to classify images (step 202).

[23] As described above, the machine learning model is a model, e.g., a deep convolutional neural network, that is configured to receive an input image and to process the input image to generate a predicted point in an embedding space in accordance with values of the parameters of the model.

[24] The training data includes multiple training images and respective label data for each of the training images. The label data for a given training image identifies one or more object categories from the set of object categories to which one or more objects depicted in the training image belong. That is, the label data associates the training image with one or more of the object categories.

[25] The system determines label embeddings for the object categories in the set of object categories (step 204). Once the embeddings have been generated, the distance in the embedding space between the numeric embeddings of any two object categories reflects a degree of visual co-occurrence of the two object categories in the training images. In particular, the degree of visual co-occurrence is based on a relative frequency with which the same training image in the training data includes one or more objects that collectively belong to both of the two object categories, i.e., the relative frequency with which the label data for a training image associates both of the object categories with the training image.

[26] To determine the label embeddings for the object categories, the system determines a respective pointwise mutual information measure between each possible pair of object categories in the set of object categories as measured in the training data. For example, for a given pair of object categories, the pointwise mutual information measure can be the logarithm of the ratio between (i) the probability that a training image includes one or more objects that collectively belong to both of the two object categories and (ii) the product of the probability that a training image includes one or more objects that belong to the first object category in the pair and the probability that a training image includes one or more objects that belong to the second object category in the pair.

[27] The system then constructs a matrix of the pointwise mutual information measures. In particular, the system constructs the matrix such that for all i and j, the entry (i,j) of the matrix is the pointwise mutual information measure between the category that is in position i in an order of the object categories and the category that is in position j in the order.

[28] The system then performs an eigen-decomposition of the matrix of pointwise mutual information measures to determine an embedding matrix. For example, the system can decompose, e.g., via singular value decomposition, the matrix of pointwise mutual information measures *PMI* into a matrix product of matrices that satisfies:

$$PMI = U \cdot \Sigma \cdot U^t,$$

where Σ is a diagonal matrix that has eigenvalues ranked from most significant to least significant along the diagonal. The embedding matrix E can then satisfy:

$$E = U \cdot \Sigma^{1/2}.$$

[29] The system then determines the numeric embeddings from the rows of the embedding matrix. In particular, the system restricts the embedding matrix to its first k columns to generate a restricted embedding matrix and then uses the rows of the restricted embedding matrix as the numeric embeddings, i.e., so that row i of the restricted embedding matrix is the numeric embedding for the category that is in position i in the order.

[30] The system trains the machine learning model on the training data to determine trained values of the model parameters from initial values of the model parameters (step 206).

[31] In particular, for each of the training images, the system processes the training image using the machine learning model in accordance with current values of the parameters of the machine learning model to generate a predicted point in the embedding space for the training image.

[32] The system then determines an adjustment to the current values of the parameters that reduces the distance, e.g., according to cosine proximity, between the predicted point in the embedding space and the numeric embeddings of the object categories identified in the label data for the training image, e.g., using a gradient descent based machine learning training technique, e.g., RMSprop. When there is more than one object category identified in the label data for the training image, the system can determine a combined embedding from the numeric embeddings of the object categories identified in the label data for the training image and then adjust the current values of the parameters to reduce the cosine proximity between the combined embedding and the predicted point in the embedding space for the training image using the machine learning training technique. In some implementations, the system determines the combined embedding by summing the numeric embeddings of the object categories identified in the label data for the training image.

[33] Once the system has trained the machine learning model, the system can use the numeric embeddings and the trained parameter values to classify new images using the trained model.

[34] FIG. 3 is a flow diagram of an example process 300 for classifying a new image using a trained machine learning model.

[35] For convenience, the process 300 will be described as being performed by a system of one or more computers located in one or more locations. For example, an image classification system, e.g., the image classification system 100 of FIG. 1, appropriately programmed in accordance with this specification, can perform the process 300.

[36] The system maintains label embedding data (step 302). The label embedding data is data that maps each object category in the set of object categories to a respective numeric embedding of the object category in an embedding space. As described above, the distance in the embedding space, e.g., according to cosine proximity or another appropriate distance metric, between the numeric embeddings for any two object categories reflects a degree of visual co-occurrence of the two object categories in images, e.g., in images that were used to train the machine learning model. In particular, the numeric embeddings for two object categories that co-occur more frequently will generally be closer in the embedding space than the numeric embeddings for two object categories that co-occur relatively less frequently.

[37] The system receives a new image to be classified (step 304).

[38] The system processes the new image using the trained machine learning model to determine a predicted point in the embedding space (step 306). As described above, the machine learning model has been configured through training to receive the new image and to process the new image to generate the predicted point in accordance with trained values of the parameters of the model.

[39] The system determines one or more numeric embeddings from the numeric embeddings for the object categories to the predicted point according to an appropriate distance metric, e.g., cosine proximity (step 308). In some implementations, the system determines a predetermined number of numeric embeddings that are closest to the predicted point in the embedding space. In some other implementations, the system identifies each numeric embedding that is closer than a threshold distance to the predicted point in the embedding space.

[40] The system classifies the new image as including images of one or more objects that belong to the object categories represented by the one or more closest numeric embeddings (step 310). Once the new image has been classified, the system can provide data identifying

the object categories for presentation to a user, e.g., ranked according to how close the corresponding embeddings were to the predicted point, store data identifying the object categories for later use, or provide the data identifying the object categories to an external system for use for some immediate purpose.

[41] This specification uses the term "configured" in connection with systems and computer program components. For a system of one or more computers to be configured to perform particular operations or actions means that the system has installed on it software, firmware, hardware, or a combination of them that in operation cause the system to perform the operations or actions. For one or more computer programs to be configured to perform particular operations or actions means that the one or more programs include instructions that, when executed by data processing apparatus, cause the apparatus to perform the operations or actions.

[42] Embodiments of the subject matter and the functional operations described in this specification can be implemented in digital electronic circuitry, in tangibly-embodied computer software or firmware, in computer hardware, including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Embodiments of the subject matter described in this specification can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions encoded on a tangible non transitory storage medium for execution by, or to control the operation of, data processing apparatus. The computer storage medium can be a machine-readable storage device, a machine-readable storage substrate, a random or serial access memory device, or a combination of one or more of them. Alternatively or in addition, the program instructions can be encoded on an artificially generated propagated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus for execution by a data processing apparatus.

[43] The term "data processing apparatus" refers to data processing hardware and encompasses all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can also be, or further include, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application specific integrated circuit). The apparatus can optionally include, in addition to hardware, code that creates an execution environment for computer programs, e.g., code that constitutes processor firmware, a

protocol stack, a database management system, an operating system, or a combination of one or more of them.

[44] A computer program, which may also be referred to or described as a program, software, a software application, an app, a module, a software module, a script, or code, can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages; and it can be deployed in any form, including as a stand alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data, e.g., one or more scripts stored in a markup language document, in a single file dedicated to the program in question, or in multiple coordinated files, e.g., files that store one or more modules, sub programs, or portions of code. A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a data communication network.

[45] In this specification, the term "database" is used broadly to refer to any collection of data: the data does not need to be structured in any particular way, or structured at all, and it can be stored on storage devices in one or more locations. Thus, for example, the index database can include multiple collections of data, each of which may be organized and accessed differently.

[46] Similarly, in this specification the term "engine" is used broadly to refer to a software-based system, subsystem, or process that is programmed to perform one or more specific functions. Generally, an engine will be implemented as one or more software modules or components, installed on one or more computers in one or more locations. In some cases, one or more computers will be dedicated to a particular engine; in other cases, multiple engines can be installed and running on the same computer or computers.

[47] The processes and logic flows described in this specification can be performed by one or more programmable computers executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by special purpose logic circuitry, e.g., an FPGA or an ASIC, or by a combination of special purpose logic circuitry and one or more programmed computers.

[48] Computers suitable for the execution of a computer program can be based on general or special purpose microprocessors or both, or any other kind of central processing unit. Generally, a central processing unit will receive instructions and data from a read only memory or a random access memory or both. The essential elements of a computer are a

central processing unit for performing or executing instructions and one or more memory devices for storing instructions and data. The central processing unit and the memory can be supplemented by, or incorporated in, special purpose logic circuitry. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, e.g., a mobile telephone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a Global Positioning System (GPS) receiver, or a portable storage device, e.g., a universal serial bus (USB) flash drive, to name just a few.

[49] Computer readable media suitable for storing computer program instructions and data include all forms of non volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto optical disks; and CD ROM and DVD-ROM disks.

[50] To provide for interaction with a user, embodiments of the subject matter described in this specification can be implemented on a computer having a display device, e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's device in response to requests received from the web browser. Also, a computer can interact with a user by sending text messages or other forms of message to a personal device, e.g., a smartphone that is running a messaging application, and receiving responsive messages from the user in return.

[51] Data processing apparatus for implementing machine learning models can also include, for example, special-purpose hardware accelerator units for processing common and compute-intensive parts of machine learning training or production, i.e., inference, workloads.

[52] Machine learning models can be implemented and deployed using a machine learning framework, .e.g., a TensorFlow framework, a Microsoft Cognitive Toolkit framework, an Apache Singa framework, or an Apache MXNet framework.

[53] Embodiments of the subject matter described in this specification can be implemented in a computing system that includes a back end component, e.g., as a data server, or that includes a middleware component, e.g., an application server, or that includes a front end component, e.g., a client computer having a graphical user interface, a web browser, or an app through which a user can interact with an implementation of the subject matter described in this specification, or any combination of one or more such back end, middleware, or front end components. The components of the system can be interconnected by any form or medium of digital data communication, e.g., a communication network. Examples of communication networks include a local area network (LAN) and a wide area network (WAN), e.g., the Internet.

[54] The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other. In some embodiments, a server transmits data, e.g., an HTML page, to a user device, e.g., for purposes of displaying data to and receiving user input from a user interacting with the device, which acts as a client. Data generated at the user device, e.g., a result of the user interaction, can be received at the server from the device.

[55] While this specification contains many specific implementation details, these should not be construed as limitations on the scope of any invention or on the scope of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments of particular inventions. Certain features that are described in this specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially be claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[56] Similarly, while operations are depicted in the drawings and recited in the claims in a particular order, this should not be understood as requiring that such operations be performed

in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system modules and components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

[57] Particular embodiments of the subject matter have been described. Other embodiments are within the scope of the following claims. For example, the actions recited in the claims can be performed in a different order and still achieve desirable results. As one example, the processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In some cases, multitasking and parallel processing may be advantageous.

WHAT IS CLAIMED IS:

1. A method comprising:

obtaining training data for training a machine learning model having a plurality of parameters,

wherein the machine learning model is configured to process input images to generate, for each input image, a predicted point in an embedding space, and

wherein the training data comprises a plurality of training images and, for each training image, label data that identifies one or more object categories from a set of object categories to which one or more objects depicted in the training image belong;

determining, from the label data for the training images in the training data, a respective numeric embedding in the embedding space of each of the object categories in the set of object categories, wherein a distance in the embedding space between the numeric embeddings of any two object categories reflects a degree of visual co-occurrence of the two object categories in the training images; and

training the machine learning model on the training data, comprising, for each of the training images:

processing the training image using the machine learning model in accordance with current values of the parameters to generate a predicted point in the embedding space for the training image; and

adjusting the current values of the parameters to reduce a distance between the predicted point in the embedding space and the numeric embeddings of the object categories identified in the label data for the training image.

2. The method of claim 1, wherein the degree of visual co-occurrence is based on a relative frequency with which a same training image in the training data includes one or more objects that collectively belong to both of the two object categories.

3. The method of any one of claims 1 or 2, wherein determining the respective embedding of each of the object categories comprises:

determining a respective pointwise mutual information measure between each possible pair of object categories in the set of object categories as measured in the training data;

constructing a matrix of the pointwise mutual information measures;

performing an eigen-decomposition of the matrix of pointwise mutual information

measures to determine an embedding matrix; and

determining the numeric embeddings from the rows of the embedding matrix.

4. The method of claim 3, wherein determining the numeric embeddings from the rows of the embedding matrix comprises:

restricting the embedding matrix to its first k columns to generate a restricted embedding matrix; and

using the rows of the restricted embedding matrix as the numeric embeddings.

5. The method any one of claims 3 or 4, wherein performing an eigen-decomposition of the matrix of pointwise mutual information measures to determine an embedding matrix comprises:

decomposing the matrix of pointwise mutual information measures PMI into a matrix product of matrices that satisfies:

$$PMI = U \cdot \Sigma \cdot U^t,$$

where Σ has eigenvalues ranked from most significant to least significant in a main diagonal, wherein the embedding matrix E satisfies:

$$E = U \cdot \Sigma^{1/2}.$$

6. The method of any one of claims 1-5, wherein adjusting the current values of the parameters comprises:

determining a combined embedding from the numeric embeddings of the object categories identified in the label data for the training image; and

adjusting the current values of the parameters to reduce a cosine proximity between the combined embedding and the predicted point in the embedding space for the training image.

7. The method of claim 6, wherein determining the combined embedding comprises summing the numeric embeddings of the object categories identified in the label data for the training image.

8. The method of any one of claims 1-7, wherein the machine learning model is a deep convolutional neural network.

9. A method comprising:

maintaining data that maps each object category in a set of object categories to a respective numeric embedding of the object category in an embedding space, wherein a distance in the embedding space between the numeric embeddings for any two object categories reflects a degree of visual co-occurrence of the two object categories in images;

receiving an input image;

processing the input image using a machine learning model, wherein the machine learning model has been configured to process the input image to generate a predicted point in the embedding space;

determining, from the maintained data, one or more numeric embeddings that are closest to the predicted point in the embedding space; and

classifying the input image as including images of one or more objects that belong to the object categories represented by the one or more numeric embeddings.

10. The method of claim 9, wherein the machine learning model is a deep convolutional neural network.

11. The method of any one of claims 9 or 10, wherein the degree of visual co-occurrence is based on a relative frequency with which a same training image in training data used to train the machine learning model includes one or more objects that collectively belong to both of the two object categories.

12. The method of any one of claims 9-11, wherein determining, from the maintained data, one or more numeric embeddings that are closest to the predicted point in the embedding space comprises:

determining a predetermined number of numeric embeddings that are closest to the predicted point in the embedding space.

13. The method of any one of claims 9-11, wherein determining, from the maintained data, one or more numeric embeddings that are closest to the predicted point in the embedding space comprises:

identifying each numeric embedding that is closer than a threshold distance to the predicted point in the embedding space.

14. The method of any one of claim 9-13, wherein determining, from the maintained data, one or more numeric embeddings that are closest to the predicted point in the embedding space comprises:

using cosine proximity to determine the one or more numeric embeddings that are closest to the predicted point.

15. A system comprising one or more computers and one or more storage devices storing instructions that are operable, when executed by the one or more computers, to cause the one or more computers to perform the operations of the respective method of any one of claims 1-14.

16. A computer storage medium encoded with instructions that, when executed by one or more computers, cause the one or more computers to perform the operations of the respective method of any one of claims 1-14.

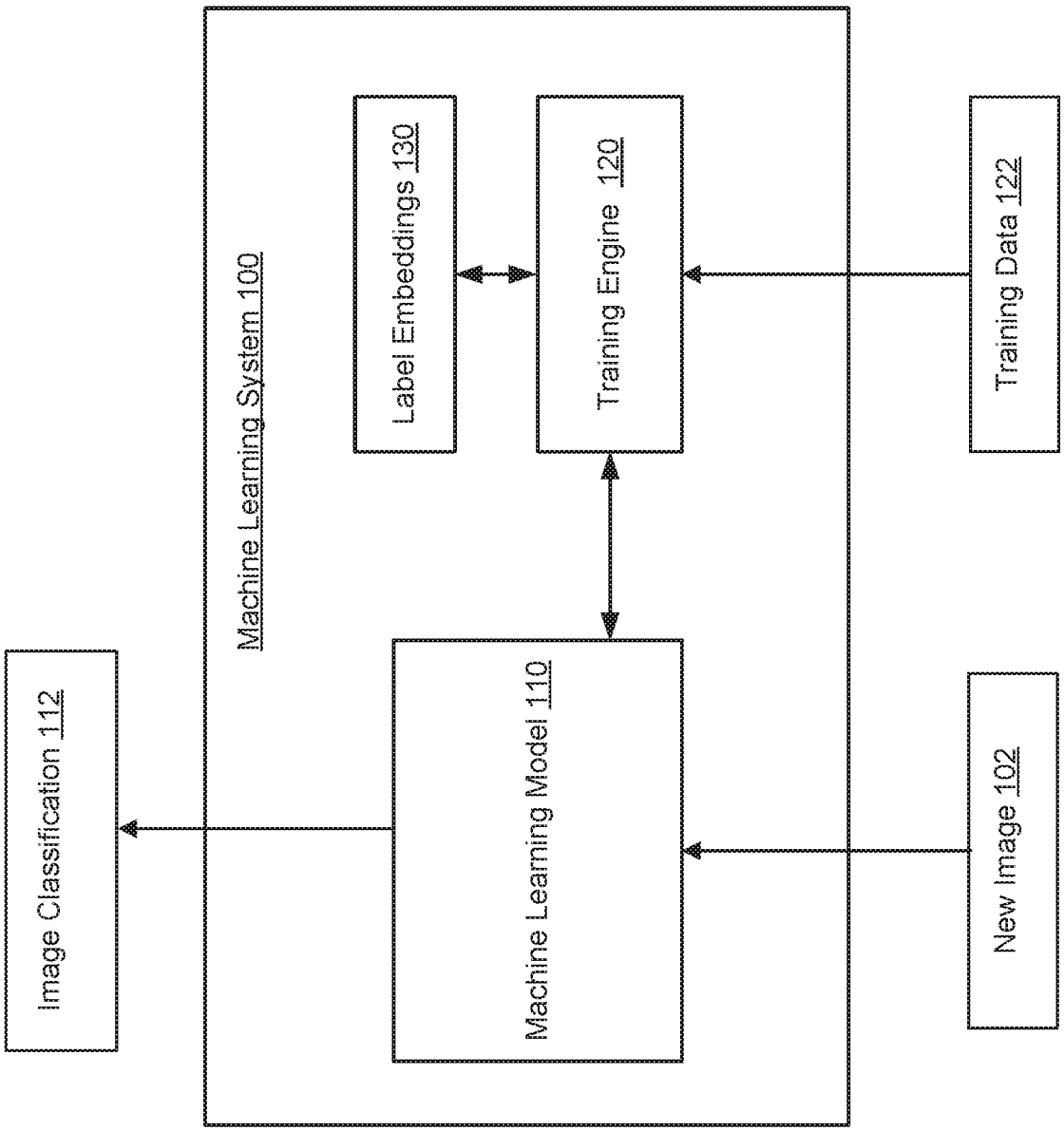
Machine Learning System 100
Label Embeddings 130
Training Engine 120
Training Data 122
Image Classification 112
Machine Learning Model 110
New Image 102

FIG. 1

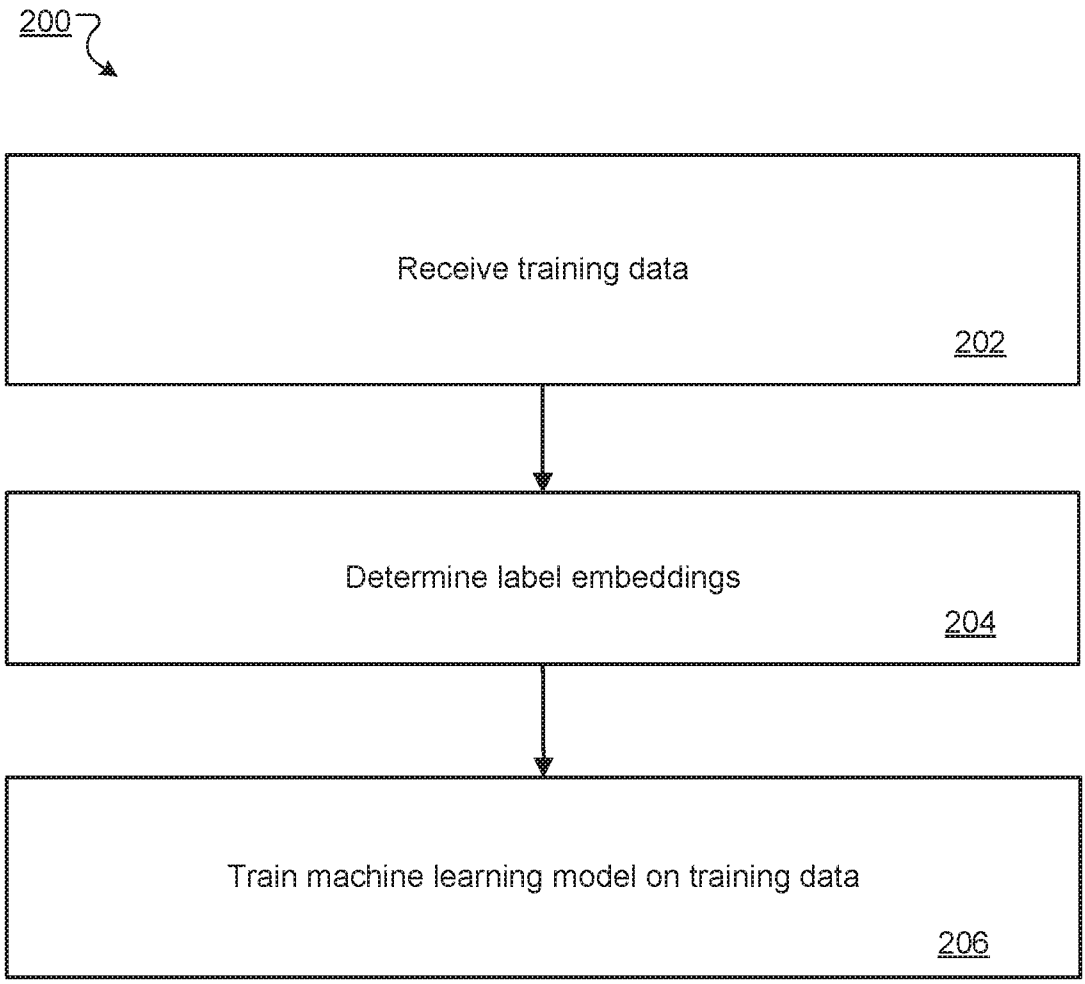
200
Receive training data
202
Determine label embeddings
204
Train machine learning model on training data
206

FIG. 2

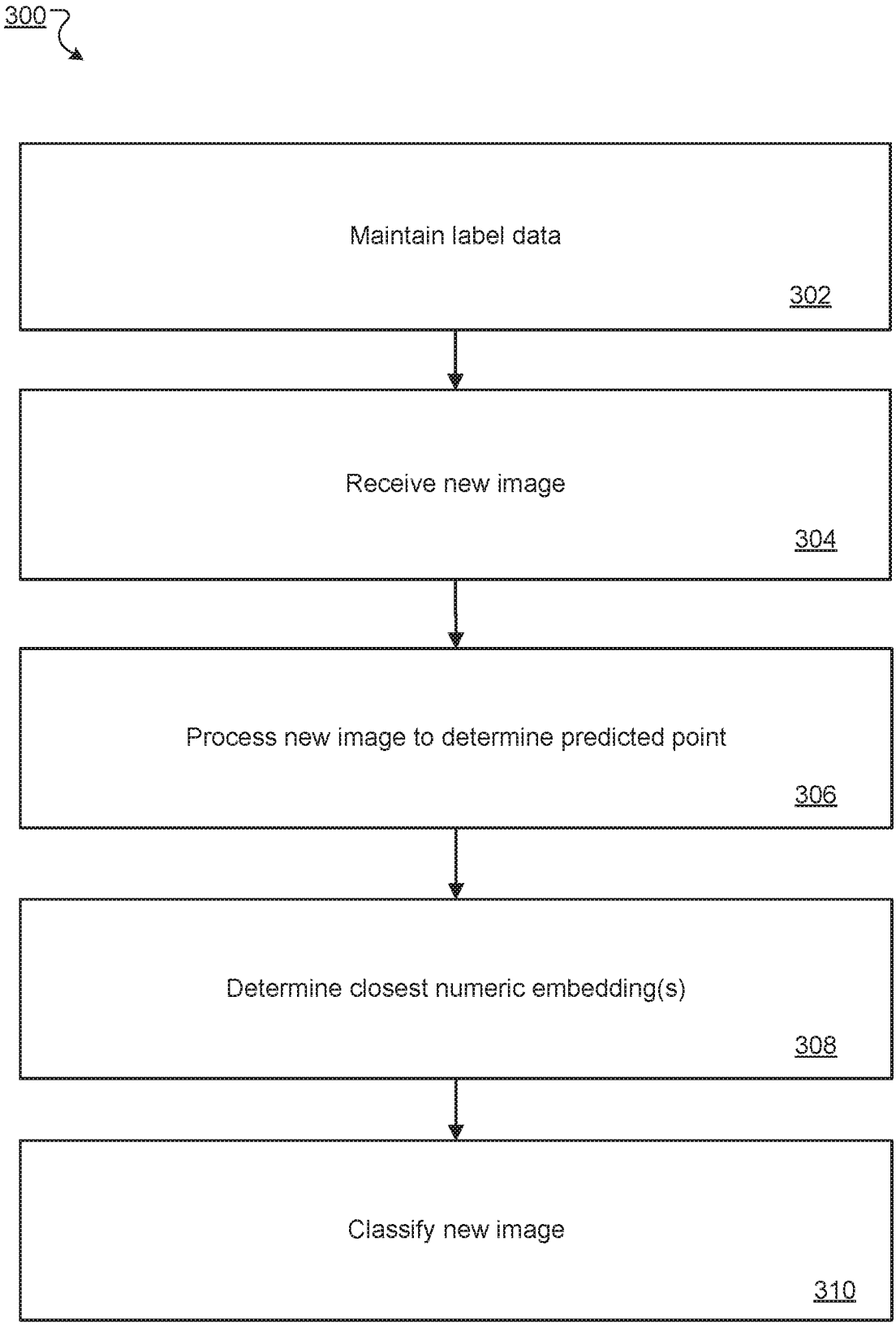
300
Maintain label data
302
Receive new image
304
Process new image to determine predicted point
306
Determine closest numeric embedding(s)
308
Classify new image
310

FIG. 3

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2017/042235

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F17/30 G06N3/04 G06N3/063 G06N3/08 G06K9/66
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06N G06K G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WO 2016/100717 A1 (GOOGLE INC [US]) 23 June 2016 (2016-06-23) paragraph [0004] - paragraph [0016]; figures 1-3 paragraph [0023] - paragraph [0047]	1-8,15, 16
X	US 2015/178383 A1 (CORRADO GREGORY SEAN [US] ET AL) 25 June 2015 (2015-06-25)	9-14
Y	paragraph [0012] - paragraph [0042]; figures 1-3	1-8,15, 16
A	US 9 141 916 B1 (CORRADO GREGORY S [US] ET AL) 22 September 2015 (2015-09-22) the whole document	1-16

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
30 November 2017	07/12/2017

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Bowler, Alyssa

3

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2017/042235

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2016100717 A1	23-06-2016	EP 3234871 A1	25-10-2017
		US 2016180151 A1	23-06-2016
		WO 2016100717 A1	23-06-2016
US 2015178383 A1	25-06-2015	NONE	
US 9141916 B1	22-09-2015	US 9141916 B1	22-09-2015
		US 9514404 B1	06-12-2016