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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
09 January 2020 (09.01.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/008108 A1

(51) International Patent Classification:
G06N 3/08 (2006.01) *H04N 19/126* (2014.01)
G06N 3/04 (2006.01) *G06T 9/00* (2006.01)

(21) International Application Number:
PCT/FI2019/050495

(22) International Filing Date:
25 June 2019 (25.06.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
20185624 05 July 2018 (05.07.2018) FI

(71) Applicant: **NOKIA TECHNOLOGIES OY** [FI/FI]; Karaportti 3, 02610 Espoo (FI).

(72) Inventors: **AYTEKIN, Caglar**; Riihuhdankatu 13B 6, 33580 Tampere (FI). **CRICRI, Francesco**; Massunkuja 1 A 14, 33100 Tampere (FI).

(74) Agent: **NOKIA TECHNOLOGIES OY** et al.; Ari Aarnio, IPR Department, Karakaari 7, 02610 Espoo (FI).

(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.

(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))*

(54) Title: AN APPARATUS, A METHOD AND A COMPUTER PROGRAM FOR TRAINING A NEURAL NETWORK

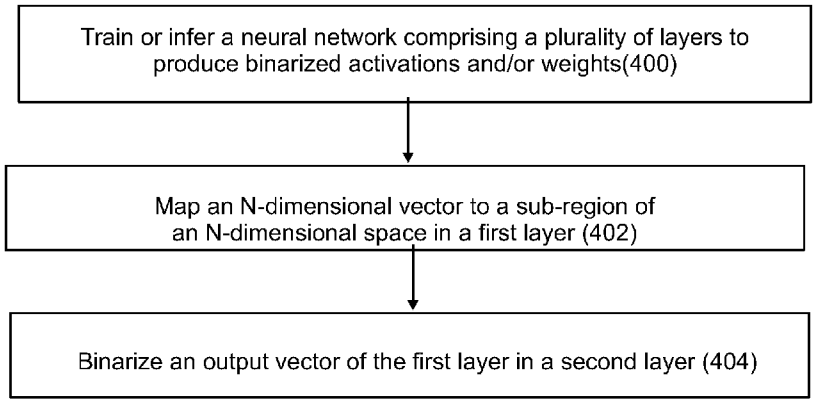

Fig. 4

(57) Abstract: A method comprising: training or inferencing a neural network comprising a plurality of layers to produce binarized activations and/or weights (400), said training or inferencing comprising: mapping an N-dimensional vector to a sub-region of an N-dimensional space in a first layer (402); and binarizing an output vector of the first layer in a second layer (404).

WO 2020/008108 A1

AN APPARATUS, A METHOD AND A COMPUTER PROGRAM FOR TRAINING A NEURAL NETWORK

TECHNICAL FIELD

[0001] The present invention relates to an apparatus, a method and a computer program for running a neural network.

BACKGROUND

[0002] Recently, the development of various artificial neural network (NN) techniques, especially the ones related to deep learning, has enabled to learn algorithms for several tasks from the raw data, which algorithms may outperform algorithms which have been developed for many years using non-learning based methods.

[0003] Binarization in neural networks has many applications such as weight binarization of neural networks, where the aim is to have neural networks of very small size. Another application is binarization of activations, i.e., binarization of the output of one or more layers of a neural network. This is particularly important for applications like image compression where the codes are aimed to be quantized/binarized for high compression rates.

[0004] Training a neural network to produce binarized activations is challenging. There are two widely used binarization approximations for neural networks; binarization as a noise, where the binarization is considered as a noise process on top of the real-valued activation, and binarization as a random variable, where the binarization is considered as a random variable having a Bernoulli distribution. Both of these can be used at the training stage of a neural network.

[0005] Both of the above approaches usually assume each i-th activation to be the output of a sigmoid activation function so that it is in [0,1] interval. This is usually needed in order to provide reasonable input to binarization process. However, when using the activations which are outputs of sigmoid function, the problem of vanishing gradients is easily encountered, wherein the vanishingly small gradients obtained from sigmoid function activations may hinder or even completely stop the neural network from further training. Moreover, the initial approximation of binarization in both of the above approaches easily becomes inaccurate, i.e. the amount of noise to be added during training may be significant.

SUMMARY

[0006] Now in order to at least alleviate the above problems, an improved method for training and running a neural network for video enhancements is introduced herein.

[0007] A method according to a first aspect comprises training or inferring a neural network comprising a plurality of layers to produce binarized activations and/or weights, said training or inferring comprising: mapping an N-dimensional vector to a sub-region of an N-dimensional space in a first layer; and binarizing an output vector of the first layer in a second layer.

[0008] According to an embodiment, the sub-region of the N-dimensional space comprises a surface at the N-dimensional space.

[0009] According to an embodiment, the surface comprises a hypersphere having radius of 0.5 and centre point located at 0.5 x **1**, wherein **1** is an N-dimensional vector of ones .

[0010] According to an embodiment, the method further comprises calculating the mapping by

$f(x) = 0.5 + 0.5\frac{x}{\|x\|_2}$, where x is N-dimensional activation vector.

[0011] According to an embodiment, the method further comprises dividing the N-dimensional input vector into pairs of 2-dimensional activations; and mapping each 2-dimensional activation to a circle that intersects corners of a square having side length of one.

[0012] According to an embodiment, the method further comprises calculating the mapping of the 2-dimensional activation z by

$g(z_i) = 0.5 + \frac{\sqrt{2}}{2}\frac{z_i}{\|z_i\|_2}$, where x is N-dimensional activation vector.

[0013] According to an embodiment, each pair of the N-dimensional activation vectors is created from neighbouring/adjacent activations x_i and x_{i+1} within the activation vector.

[0014] According to an embodiment, upon training the neural network, the binarization of the output vector of the first layer comprises adding noise, and upon inferring the neural network, the binarization of the output vector of the first layer comprises a binarization operation.

[0015] A second aspect relates to an apparatus comprising: a plurality of neural network layers comprising: a first layer configured to map an N-dimensional vector to a sub-region of an N-dimensional space; and a second layer configured to binarize an output vector of the first layer.

[0016] The further aspects relate to apparatuses and computer readable storage media stored with code thereon, which are arranged to carry out the above methods and one or more of the embodiments related thereto.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] For better understanding of the present invention, reference will now be made by way of example to the accompanying drawings in which:

[0018] Figure 1 shows schematically an electronic device employing embodiments of the invention;

[0019] Figure 2 shows schematically a user equipment suitable for employing embodiments of the invention;

[0020] Figure 3 further shows schematically electronic devices employing embodiments of the invention connected using wireless and wired network connections;

[0021] Figure 4 shows a flow chart of a method for training a neural network according to an embodiment of the invention;

[0022] Figures 5a, 5b and 5c show simplified examples of activation mappings according to embodiments of the invention;

[0023] Figure 6 shows a simplified example of mapping activations pairwise on a sphere according to an embodiment of the invention; and

[0024] Figures 7a and 7b illustrate the operation in a training phase and in an inference phase, correspondingly, according to some embodiments of the invention.

DETAILED DESCRIPTON OF SOME EXAMPLE EMBODIMENTS

[0025] In the following, several embodiments will be described in the context of encoding and decoding visual data, such as video frames. It is to be noted, however, the embodiments are not limited to processing of visual data, but the different embodiments have applications in any environment where data can be streamed and compressed. Thus, applications including but not limited to, for example, streaming of speech or other audio data can benefit from the use of the embodiments.

[0026] The following describes in further detail suitable apparatus and possible mechanisms for running a neural network according to embodiments. In this regard reference is first made to Figures 1 and 2, where Figure 1 shows an example block diagram of an apparatus 50. The apparatus may be an Internet of Things (IoT) apparatus configured to perform various functions, such as for example, gathering information by one or more

sensors, receiving or transmitting information, analyzing information gathered or received by the apparatus, or the like. The apparatus may comprise a video coding system, which may incorporate a codec. Figure 2 shows a layout of an apparatus according to an example embodiment. The elements of Figs. 1 and 2 will be explained next.

[0027] The electronic device 50 may for example be a mobile terminal or user equipment of a wireless communication system, a sensor device, a tag, or other lower power device. However, it would be appreciated that embodiments of the invention may be implemented within any electronic device or apparatus which may process data by neural networks.

[0028] The apparatus 50 may comprise a housing 30 for incorporating and protecting the device. The apparatus 50 further may comprise a display 32 in the form of a liquid crystal display. In other embodiments of the invention the display may be any suitable display technology suitable to display an image or video. The apparatus 50 may further comprise a keypad 34. In other embodiments of the invention any suitable data or user interface mechanism may be employed. For example the user interface may be implemented as a virtual keyboard or data entry system as part of a touch-sensitive display.

[0029] The apparatus may comprise a microphone 36 or any suitable audio input which may be a digital or analogue signal input. The apparatus 50 may further comprise an audio output device which in embodiments of the invention may be any one of: an earpiece 38, speaker, or an analogue audio or digital audio output connection. The apparatus 50 may also comprise a battery (or in other embodiments of the invention the device may be powered by any suitable mobile energy device such as solar cell, fuel cell or clockwork generator). The apparatus may further comprise a camera capable of recording or capturing images and/or video. The apparatus 50 may further comprise an infrared port for short range line of sight communication to other devices. In other embodiments the apparatus 50 may further comprise any suitable short range communication solution such as for example a Bluetooth wireless connection or a USB/firewire wired connection.

[0030] The apparatus 50 may comprise a controller 56, processor or processor circuitry for controlling the apparatus 50. The controller 56 may be connected to memory 58 which in embodiments of the invention may store both data in the form of image and audio data and/or may also store instructions for implementation on the controller 56. The controller 56 may further be connected to codec circuitry 54 suitable for carrying out coding and/or decoding of audio and/or video data or assisting in coding and/or decoding carried out by the controller.

[0031] The apparatus 50 may further comprise a card reader 48 and a smart card 46, for example a UICC and UICC reader for providing user information and being suitable for providing authentication information for authentication and authorization of the user at a network.

[0032] The apparatus 50 may comprise radio interface circuitry 52 connected to the controller and suitable for generating wireless communication signals for example for communication with a cellular communications network, a wireless communications system or a wireless local area network. The apparatus 50 may further comprise an antenna 44 connected to the radio interface circuitry 52 for transmitting radio frequency signals generated at the radio interface circuitry 52 to other apparatus(es) and/or for receiving radio frequency signals from other apparatus(es).

[0033] The apparatus 50 may comprise a camera capable of recording or detecting individual frames which are then passed to the codec 54 or the controller for processing. The apparatus may receive the video image data for processing from another device prior to transmission and/or storage. The apparatus 50 may also receive either wirelessly or by a wired connection the image for coding/decoding. The structural elements of apparatus 50 described above represent examples of means for performing a corresponding function.

[0034] With respect to Figure 3, an example of a system within which embodiments of the present invention can be utilized is shown. The system 10 comprises multiple communication devices which can communicate through one or more networks. The system 10 may comprise any combination of wired or wireless networks including, but not limited to a wireless cellular telephone network (such as a GSM, UMTS, CDMA, 4G, 5G network etc.), a wireless local area network (WLAN) such as defined by any of the IEEE 802.x standards, a Bluetooth personal area network, an Ethernet local area network, a token ring local area network, a wide area network, and the Internet.

[0035] The system 10 may include both wired and wireless communication devices and/or apparatus 50 suitable for implementing embodiments of the invention.

[0036] For example, the system shown in Figure 3 shows a mobile telephone network 11 and a representation of the internet 28. Connectivity to the internet 28 may include, but is not limited to, long range wireless connections, short range wireless connections, and various wired connections including, but not limited to, telephone lines, cable lines, power lines, and similar communication pathways.

[0037] The example communication devices shown in the system 10 may include, but are not limited to, an electronic device or apparatus 50, a combination of a personal digital

assistant (PDA) and a mobile telephone 14, a PDA 16, an integrated messaging device (IMD) 18, a desktop computer 20, a notebook computer 22. The apparatus 50 may be stationary or mobile when carried by an individual who is moving. The apparatus 50 may also be located in a mode of transport including, but not limited to, a car, a truck, a taxi, a bus, a train, a boat, an airplane, a bicycle, a motorcycle or any similar suitable mode of transport.

[0038] The embodiments may also be implemented in a set-top box; i.e. a digital TV receiver, which may/may not have a display or wireless capabilities, in tablets or (laptop) personal computers (PC), which have hardware and/or software to process neural network data, in various operating systems, and in chipsets, processors, DSPs and/or embedded systems offering hardware/software based coding.

[0039] Some or further apparatus may send and receive calls and messages and communicate with service providers through a wireless connection 25 to a base station 24. The base station 24 may be connected to a network server 26 that allows communication between the mobile telephone network 11 and the internet 28. The system may include additional communication devices and communication devices of various types.

[0040] The communication devices may communicate using various transmission technologies including, but not limited to, code division multiple access (CDMA), global systems for mobile communications (GSM), universal mobile telecommunications system (UMTS), time divisional multiple access (TDMA), frequency division multiple access (FDMA), transmission control protocol-internet protocol (TCP-IP), short messaging service (SMS), multimedia messaging service (MMS), email, instant messaging service (IMS), Bluetooth, IEEE 802.11, 3GPP Narrowband IoT and any similar wireless communication technology. A communications device involved in implementing various embodiments of the present invention may communicate using various media including, but not limited to, radio, infrared, laser, cable connections, and any suitable connection.

[0041] In telecommunications and data networks, a channel may refer either to a physical channel or to a logical channel. A physical channel may refer to a physical transmission medium such as a wire, whereas a logical channel may refer to a logical connection over a multiplexed medium, capable of conveying several logical channels. A channel may be used for conveying an information signal, for example a bitstream, from one or several senders (or transmitters) to one or several receivers.

[0042] The embodiments may also be implemented in so-called IoT devices. The Internet of Things (IoT) may be defined, for example, as an interconnection of uniquely

identifiable embedded computing devices within the existing Internet infrastructure. The convergence of various technologies has and will enable many fields of embedded systems, such as wireless sensor networks, control systems, home/building automation, etc. to be included the Internet of Things (IoT). In order to utilize Internet IoT devices are provided with an IP address as a unique identifier. IoT devices may be provided with a radio transmitter, such as WLAN or Bluetooth transmitter or a RFID tag. Alternatively, IoT devices may have access to an IP-based network via a wired network, such as an Ethernet-based network or a power-line connection (PLC).

[0043] Recently, the development of various artificial neural network (NN) techniques, especially the ones related to deep learning, has enabled to learn algorithms for several tasks from the raw data, which algorithms may outperform algorithms which have been developed for many years using traditional (non-learning based) methods.

[0044] Artificial neural networks, or simply neural networks, are parametric computation graphs comprising units and connections. The units may be arranged in successive layers, and in some neural network architectures only units in adjacent layers are connected. Each connection has an associated parameter or weight, which defines the strength of the connection. The weight gets multiplied by the incoming signal in that connection. In fully-connected layers of a feedforward neural network, each unit in a layer is connected to each unit in the following layer. So, the signal which is output by a certain unit gets multiplied by the connections connecting that unit to another unit in the following layer. The latter unit then may perform a simple operation such as a sum of the weighted signals.

[0045] Apart from fully-connected layers, there are different types of layers, such as but not limited to convolutional layers, non-linear activation layers, batch-normalization layers, and pooling layers.

[0046] The input layer receives the input data, such as images, and the output layer is task-specific and outputs an estimate of the desired data, for example a vector whose values represent a class distribution in the case of image classification. The "quality" of the neural network's output is evaluated by comparing it to ground-truth output data. The comparison may include a loss or cost function, run on the neural network's output and the ground-truth data. This comparison would then provide a "loss" or "cost" value.

[0047] The weights of the connections represent the biggest part of the learnable parameters of a neural network. Hereinafter, the terms "model" and "neural network" are

used interchangeably, as well as the weights of neural networks are sometimes referred to as learnable parameters or simply as parameters.

[0048] The parameters are learned by means of a training algorithm, where the goal is to minimize the loss value on a training dataset and on a held-out validation dataset. In order to minimize such value, the network is run on a training dataset, a loss value is computed for the whole training dataset or for part of it, and the learnable parameters are modified in order to minimize the loss value on the training dataset. However, the performance of the training is evaluated on the held-out validation dataset. The training dataset is regarded as a representative sample of the whole data. One learning approach is based on iterative local methods, where the loss on the training dataset is minimized by following the negative gradient direction. Here, the gradient is understood to be the gradient of the loss with respect to the learnable parameters of the neural network. The loss may be represented by the reconstructed prediction error. Computing the gradient on the whole training dataset may be computationally too heavy, thus learning is performed in sub-steps, where at each step a mini-batch of data is sampled and gradients are computed from the mini-batch. This is referred to as stochastic gradient descent. The gradients are usually computed by back-propagation algorithm, where errors are propagated from the output layer to the input layer, by using the chain rule for differentiation. If the loss function or some operations performed by the neural network are not differentiable, it is still possible to estimate the gradient of the loss by using policy gradient methods, such as those used in reinforcement learning. The computed gradients are then used by one of the available optimization routines (such as stochastic gradient descent, Adam, RMSprop, etc.), to compute a weight update, which is then applied to update the weights of the network. After a full pass over the training dataset, the process is repeated several times until a convergence criterion is met, usually a generalization criterion. A generalization criterion may be derived from the loss value on the held-out validation dataset, for example by stopping the training when the loss value on the held-out validation dataset is less than a certain threshold. The gradients of the loss, i.e., the gradients of the reconstructed prediction error with respect to the weights of the neural network, may be referred to as the training signal.

[0049] Training a neural network is an optimization process, but as a difference to a typical optimization where the only goal is to minimize a function, the goal of the optimization or training process in machine learning is to make the model to learn the properties of the data distribution. In other words, the goal is to learn to generalize to

previously unseen data, i.e., data which was not used for training the model. This is usually referred to as generalization. In practice, data is usually split into two (or more) sets, the training set and the validation set. The training set is used for training the network, i.e., to modify its learnable parameters to minimize the loss. The validation set is used for checking the performance of the network on data which was not used to minimize the loss, as an indication of the final performance of the model. In particular, the errors on the training set and on the validation set are monitored during the training process to understand the following issues:

- If the network is learning at all – in this case, the training set error should decrease, otherwise we are in the regime of underfitting.
- If the network is learning to generalize – in this case, also the validation set error needs to decrease and to be not too much higher than the training set error. If the training set error is low, but the validation set error is much higher than the training set error, or it does not decrease, or it even increases, the model is in the regime of overfitting. This means that the model has just memorized the training set's properties and performs well only on that set, but performs poorly on a set not used for tuning its parameters.

[0050] Binarization in neural networks has many applications such as weight binarization of neural networks, where the aim is to have neural networks of very small size. Another application is binarization of activations, i.e., binarization of the output of one or more layers of a neural network. This is particularly important for applications like image compression where the codes are aimed to be quantized/binarized for high compression rates.

[0051] Training a neural network to produce binarized activations is challenging. Since a neural network requires differentiable layers to be trained, neural network training with binarized values obtained from an actual binarization operation is not possible. This is due to the fact that binarization operation is not continuously differentiable, and the derivatives in the regions where the binarization operation is differentiable are all zeros. Therefore, the binarization has to be approximated with a continuously differentiable alternative.

[0052] Two binarization approximations for neural networks are briefly described next. These are both used at training stage.

[0053] Binarization as a Noise:

[0054] This approach considers the binarization as a noise process on top of the real-valued activation. For example, the binarization of the value 0.7 outputs the value 1. This can be considered as adding a 0.3 valued noise on top of the actual activation.

[0055] Hence, taking this consideration into account, a neural network can be trained to be robust within a certain noise interval where the actual binarization is also included. This is achieved as follows. During forward propagation of the training process, the neural network prediction is evaluated with the added noise on the activation to be binarized. So, the addition of noise during training simulates the real binarization operation.

[0056] In order to have a better simulation of binarization via additive noise, noise may be added such that the i-th activation x_i is mapped to an interval between [0,x_i] if x is closer to 0 than 1, otherwise it is mapped to an interval between [x_i,1].

[0057] Binarization as a Random Variable:

[0058] This approach considers binarization as a random variable having a Bernoulli distribution. The distribution involves a variable p such that with probability p the distribution generates number 1 and with probability (1-p), the distribution generates number 0.

[0059] In neural network training, the variable p is directly selected as the real valued activation. For example, if the i-th activation's value is 0.8, then with 0.8 probability it is mapped to 1, and with 0.2 probability it is mapped to 0.

[0060] Both of the above approaches usually assume each i-th activation to be the output of a sigmoid activation function so that it is in [0,1] interval. This is usually needed in order to provide reasonable input to binarization process. However, when using the activations which are outputs of sigmoid function, the problem of vanishing gradients is easily encountered in training neural networks with gradient-based learning methods and backpropagation. When updating the neural network's weights, the gradients obtained from sigmoid function activations may be vanishingly small, thereby preventing the weight from updating its value, or even completely stop the neural network from further training. Moreover, the initial approximation of binarization in both of the above approaches easily becomes inaccurate, i.e. the amount of noise to be added during training may be significant.

[0061] An enhanced method for providing an approximation of binarization in neural networks is introduced herein.

[0062] An example of a method, which is depicted in the flow chart of Figure 4, comprises training or inferring (400) a neural network comprising a plurality of layers to

produce binarized activations and/or weights, said training or inferring comprising: mapping (402) an N-dimensional vector to a sub-region of an N-dimensional space in a first layer; and binarizing (404) an output vector of the first layer in a second layer.

[0063] It is noted that "inferring a neural network", as used herein, refers to utilizing the neural network at inference stage (a.k.a. testing stage, or utilization stage) for a given task, e.g. image or video compression.

[0064] Thus, by mapping the activation and/or weights to only a sub-region of an N-dimensional space prior to noise addition, the noise to be simulated by the neural network is more suitable to learn binarization approximation better. Moreover, the activations do not need to be outputs of sigmoid function.

[0065] A further advantage is obtained in the case where the binarization simulation according to the embodiments is used at the end of an encoder neural network, where the encoder is part of an auto-encoder network trained end-to-end to compress data. Thus, the decoder is trained on activations and/or weights which are more suitable for approximation of binarization.

[0066] According to an embodiment, the sub-region of the N-dimensional space comprises a surface at the N-dimensional space.

[0067] Hence, by limiting the sub-regions to certain geometrical shape surfaces, the learning to approximate binarization is more effective and the noise to be simulated by the neural network is more suitable.

[0068] The embodiments are described in the following by referring to an example scenario of a neural network with M layers, wherein focus of interest is in the binarization of the activations of the L-th layer's outputs. These activations are usually in the form of a vector of real values. With the binarization as noise approach, it may be assumed that the layers from L+1 to M will learn to be robust to changes in their input due to noise.

[0069] For example, the M layers may be the layers of a neural auto-encoder. In this case, the initial L layers may be the layers of the encoder part. Thus, the binarization simulation is applied on the output activations of the encoder. In such an assumption, the amount of this noise is important. If the noise is too high, the rest of the network can be insufficient to achieve robustness.

[0070] Considering an activation from the output of a sigmoid activation function, each i-th activation of the n-dimensional activations vector that is obtained from a sigmoid layer takes values in the range [0,1]. Thus, the n-dimensional activations vector takes values inside an n-dimensional hypercube where the corner coordinates of the hypercube are

every possible n-dimensional binary combination. For example, in case n=2, the activations vector takes values within the whole area of a square whose vertices in Cartesian coordinates are (0,0), (0,1), (1,1), (1,0). This is illustrated in Figure 5a, which shows the worst case scenario for such method, where the maximum noise amount is depicted by the length of the arrow.

[0071] Consider the n-dimensional activation $[0.5, 0.5, \ldots, 0.5]$, where the vector consists of all 0.5. Then, the noise to be simulated by the network is the hyper-cube between $[0.5, 0.5, \ldots, 0.5]$ and $[1,1, \ldots, 1]$ where the latter vector is a vector of ones. This region corresponds to a large region. Expecting robustness within this entire interval is infeasible.

[0072] In order to have a better robustness, i.e. a better binarization approximation, the simulated interval should be well suited.

[0073] According to an embodiment, the surface comprises a hypersphere having radius of 0.5 and centre point located at 0.5 x **1**, wherein **1** is an N-dimensional vector of ones .

[0074] Consequently, every activation is mapped to a sphere surface that tightly fits to the hypercube; that is to the surface of a sphere having a radius of 0.5 and a center of all 0.5s: $[0.5, 0.5, \ldots, 0.5]$. This is illustrated in Figure 5b, which shows a 2D case, where the activations are mapped to the circle. For the sake of comparison, Figure 5b also shows the square area resulting from another approach, as shown in Figure 5a.

[0075] According to an embodiment, the method further comprises calculating the mapping by

$$f(x) = 0.5 + 0.5\frac{x}{\|x\|_2}, \qquad (1.)$$

where x is N-dimensional activation vector.

[0076] Thus, considering that the points to be approximated (i.e., the binary vectors) are the corners of the square (hypercube in N-dimensional case), the activations on the circle provide more suitable values when compared to the ones within the square on average. Therefore, the noise to be simulated in binarization approximation is well-suited for the sphere surface mapping according to the embodiment compared to the conventional approach. Figure 5c shows the maximum noise amount to be simulated in the sphere surface mapping according to the embodiment by the length of the arrow. Compared to the maximum noise amount of the conventional method shown in Figure 5a, it can be clearly

observed that in the sphere surface mapping according to the embodiment the amount of the noise to be simulated is much less.

[0077] According to an embodiment, the method further comprises dividing the N-dimensional input vector into pairs of 2-dimensional activations; and mapping each 2-dimensional activation to a circle that intersects corners of a square having side length of one.

[0078] Hence, in an alternative embodiment, instead of mapping the activations on a sphere surface that fits within a hypercube, the activations are mapped on surface that intersects the corners of the hypercube. Although the sphere surface mapping according to the above embodiment greatly reduces the noise to be simulated, the activations can never have binary values, in other words, the sphere in Figures 5b and 5c does not cross the square corners which are the binary vectors.

[0079] An approach to alleviate this issue is to map the activations to the surface of a sphere that crosses the hypercube corners, i.e., to the surface of a sphere which inscribes the hypercube.

[0080] To this end, the activations may be mapped to the surface of a hypersphere of center $[0.5, 0.5, \ldots, 0.5]$ and a radius of $\frac{\sqrt{N}}{2}$, where N is the dimension of the activations. As can be observed, as N grows, the radius of such a hypersphere also grows. The maximum value that a i-th item of the activation vector can have on this sphere is $0.5+\frac{\sqrt{N}}{2}$, hence for N>3, the rounding operation on this value will be larger than 1. Therefore, the activations exceeds the interval required to perform binarization with rounding operation.

[0081] According to an embodiment, the method further comprises calculating the mapping of the 2-dimensional activation z by

$$g(z_i) = 0.5 + \frac{\sqrt{2}}{2}\frac{z_i}{\|z_i\|_2}, \qquad (2.)$$

where z comprises a pair of N-dimensional activation vectors.

[0082] Thus, the embodiment provides a solution that both enables the activations to have binary values and also stay in the interval that is required to perform binarization with rounding operation.

[0083] The embodiment may be constrained such that the N is a multiple of 2. Considering that this is satisfied, the activations are separated to pairs such that there will be N/2 pairs of 2-dimensional activations. Then, every pair of 2-d activations are mapped to a circle that intersects the corners of the square of side-length of 1. Such pairwise sphere

process is illustrated in Figure 6, where the N-dimensional vector is first separated into pairs (x_1, x_2), (x_3, x_4)…, (x_{N-1}, x_N), and then each pair is mapped to a point on the circle.

[0084] According to an embodiment, each pair of the N-dimensional activation vectors is created from neighbouring/adjacent activations x_i and x_{i+1} within the activation vector.

[0085] Separating the activations pairwise such that every consecutive pair is separated from the rest provides benefits especially in a convolutional neural network architecture utilizing the spatial correlation on the vector. For a fully connected neural network, the ordering is less important, as every neuron is connected to each other.

[0086] It should be noted that the embodiments of pairwise sphere approach are not limited to the activations of having N dimensions where N is divisible by 2 only. In fact, the embodiments can be generalized to odd number of dimensions by separating the vector into a 3-dimensional activation and $(N-3)/2$ activation pairs. Another alternative is reflective padding, i.e. padding the vector from the ends with the same activation value at the end of the vector.

[0087] It is further noted that a generalized procedure, including both the fitting-sphere and the pairwise sphere approaches, may be obtained by dividing the N dimensional vector to a matrix of MxK where (MxK=N), and performing surface mapping in one of the dimensions by

$$f(z)=c+r*z/\mathrm{norm}(z), \qquad (3.)$$

where r is radius and c is center.

[0088] From this generalized equation, the fitting-sphere approach can be reached by setting c=0.5 and r=0.5 and M=N and K=1. The pairwise sphere approach can be reached by setting c=0.5 and r=sqrt(2)/2 and M=N/2 and K=2 and the mapping dimension is of K. It is noted that the generalization according to Eq. (3) is not limited to the fitting-sphere and the pairwise sphere approaches, but it enables to derive a plurality of further variants.

[0089] For both the fitting-sphere and the pairwise sphere approaches, the training and inference strategies of the conventional binarization as a noise approach may be adopted, as shown in Figures 7a and 7b. However, as a difference to the conventional approach, no sigmoid function is used to obtain the activation vectors and the activation vectors are either mapped to a N-dimensional sphere surface with radius 0.5 and centre [0.5 ,0.5 ,…,0.5] (i.e. so-called fitting sphere approach) or first separated into pairs and each pair is mapped to a circle of radius $\sqrt{N}/2$ and of centre [0.5 ,0.5] (i.e. so-called pairwise sphere approach).

[0090] According to an embodiment, upon training the neural network, the binarization of the output vector of the first layer comprises adding noise, and upon inferring the neural network, the binarization of the output vector of the first layer comprises a binarization operation.

[0091] Thus, the binarization as referred to at training stage is actually an approximation of the effect of a real binarization, achieved by adding noise or by the random variable method. The binarization in the case of inference stage refers to actual binarization.

[0092] Consequently, during training forward pass, after one of the mappings (fitting or pairwise sphere surface), the noise is added to the activations, and during backward pass the neural networks operates on real values.

[0093] In inference phase, after applying mapping according to either the fitting-sphere or the pairwise sphere approach, the activations are binarized by rounding operation.

[0094] It is noted that while the above embodiments have been described in connection with activation binarization, the embodiments are equally applicable to weight binarization, as well.

[0095] In general, the various embodiments of the invention may be implemented in hardware or special purpose circuits, software, logic or any combination thereof. For example, some aspects may be implemented in hardware, while other aspects may be implemented in firmware or software which may be executed by a controller, microprocessor or other computing device, although the invention is not limited thereto. While various aspects of the invention may be illustrated and described as block diagrams, flow charts, or using some other pictorial representation, it is well understood that these blocks, apparatus, systems, techniques or methods described herein may be implemented in, as non-limiting examples, hardware, software, firmware, special purpose circuits or logic, general purpose hardware or controller or other computing devices, or some combination thereof.

[0096] The embodiments of this invention may be implemented by computer software executable by a data processor of the mobile device, such as in the processor entity, or by hardware, or by a combination of software and hardware. Further in this regard it should be noted that any blocks of the logic flow as in the Figures may represent program steps, or interconnected logic circuits, blocks and functions, or a combination of program steps and logic circuits, blocks and functions. The software may be stored on such physical media as memory chips, or memory blocks implemented within the processor, magnetic media such

as hard disk or floppy disks, and optical media such as for example DVD and the data variants thereof, CD.

[0097] The memory may be of any type suitable to the local technical environment and may be implemented using any suitable data storage technology, such as semiconductor-based memory devices, magnetic memory devices and systems, optical memory devices and systems, fixed memory and removable memory. The data processors may be of any type suitable to the local technical environment, and may include one or more of general purpose computers, special purpose computers, microprocessors, digital signal processors (DSPs) and processors based on multi-core processor architecture, as non-limiting examples.

[0098] Embodiments of the inventions may be practiced in various components such as integrated circuit modules. The design of integrated circuits is by and large a highly automated process. Complex and powerful software tools are available for converting a logic level design into a semiconductor circuit design ready to be etched and formed on a semiconductor substrate.

[0099] Programs, such as those provided by Synopsys, Inc. of Mountain View, California and Cadence Design, of San Jose, California automatically route conductors and locate components on a semiconductor chip using well established rules of design as well as libraries of pre-stored design modules. Once the design for a semiconductor circuit has been completed, the resultant design, in a standardized electronic format (e.g., Opus, GDSII, or the like) may be transmitted to a semiconductor fabrication facility or "fab" for fabrication.

[0100] The foregoing description has provided by way of exemplary and non-limiting examples a full and informative description of the exemplary embodiment of this invention. However, various modifications and adaptations may become apparent to those skilled in the relevant arts in view of the foregoing description, when read in conjunction with the accompanying drawings and the appended claims. However, all such and similar modifications of the teachings of this invention will still fall within the scope of this invention.

CLAIMS:

1. A method comprising

training or inferring a neural network comprising a plurality of layers to produce binarized activations and/or weights, said training or inferring comprising:

mapping an N-dimensional vector to a sub-region of an N-dimensional space in a first layer; and

binarizing an output vector of the first layer in a second layer.

2. The method according to claim 1, wherein the sub-region of the N-dimensional space comprises a surface at the N-dimensional space.

3. The method according to claim 2, wherein the surface comprises a hypersphere having radius of 0.5 and centre point located at 0.5 x **1**, wherein **1** is an N-dimensional vector of ones .

4. The method according to any preceding claim, further comprising

calculating the mapping by

$f(x) = 0.5 + 0.5\frac{x}{\|x\|_2}$, where x is N-dimensional activation vector.

5. The method according to claim 1, further comprising

dividing the N-dimensional input vector into pairs of 2-dimensional activations; and

mapping each 2-dimensional activation to a circle that intersects corners of a square having side length of one.

6. The method according to claim 5, further comprising

calculating the mapping of the 2-dimensional activation z by

$g(z_i) = 0.5 + \frac{\sqrt{2}}{2}\frac{z_i}{\|z_i\|_2}$, where x is N-dimensional activation vector.

7. The method according to claim 6, wherein each pair of the N-dimensional activation vectors is created from neighbouring/adjacent activations x_i and x_{i+1} within the activation vector.

8. The method according to any preceding claim, wherein

upon training the neural network, the binarization of the output vector of the first layer comprises adding noise, and

upon inferring the neural network, the binarization of the output vector of the first layer comprises a binarization operation.

9. An apparatus comprising:

a plurality of neural network layers comprising:

- a first layer configured to map an N-dimensional vector to a sub-region of an N-dimensional space;
- a second layer configured to binarize an output vector of the first layer.

10. The apparatus according to claim 9, wherein the sub-region of the N-dimensional space comprises a surface at the N-dimensional space.

11. The apparatus according to claim 10, wherein the surface comprises a hypersphere having radius of 0.5 and centre point located at 0.5 x **1**, wherein **1** is an N-dimensional vector of ones.

12. The apparatus according to any of claims 9 - 11, wherein the mapping is calculated by $f(x) = 0.5 + 0.5\frac{x}{\|x\|_2}$, where x is N-dimensional activation vector.

13. The apparatus according to claim 9, wherein the N-dimensional input vector is divided into pairs of 2-dimensional activations and wherein each 2-dimensional activation is mapped to a circle that intersects (at least one) corner(s) of a square having side length of one.

14. The apparatus according to claim 13, wherein mapping of the 2-dimensional activation z is calculated by $g(z_i) = 0.5 + \frac{\sqrt{2}}{2}\frac{z_i}{\|z_i\|_2}$, where x is N-dimensional activation vector.

15. The apparatus according to claim 14, wherein each pair of the N-dimensional activation vectors is created from neighbouring/adjacent activations x_i and x_{i+1} within the activation vector.

16. The apparatus of according to any of claims 9 - 15, further comprising means for processing data at the plurality of neural network layers, wherein the means comprises at least one processor and at least one memory, said at least one memory stored with code thereon, which when executed by said at least one processor, causes the performance of the apparatus.

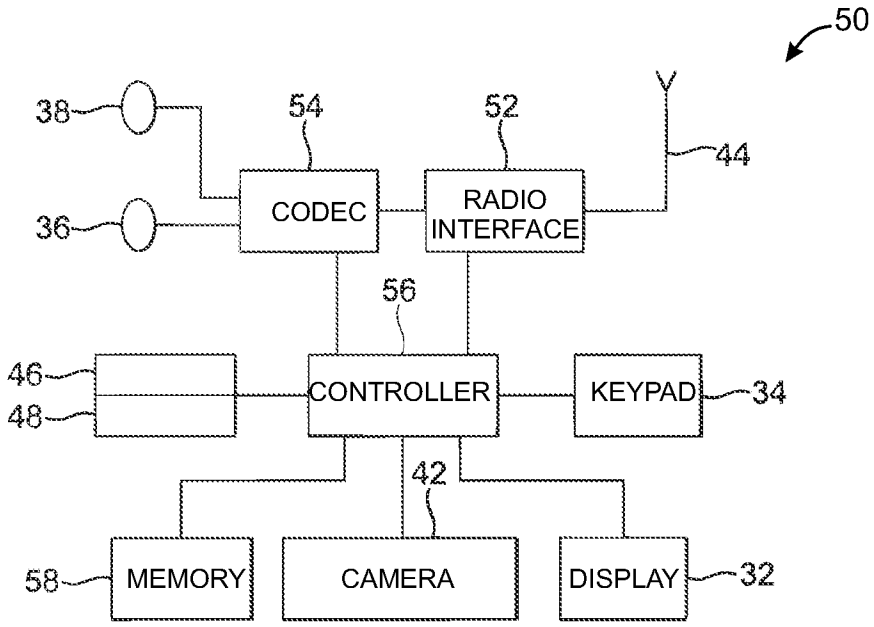
50
38
54
52
44
36
CODEC
RADIO INTERFACE
56
46
48
CONTROLLER
KEYPAD
34
42
58
MEMORY
CAMERA
DISPLAY
32

Fig. 1

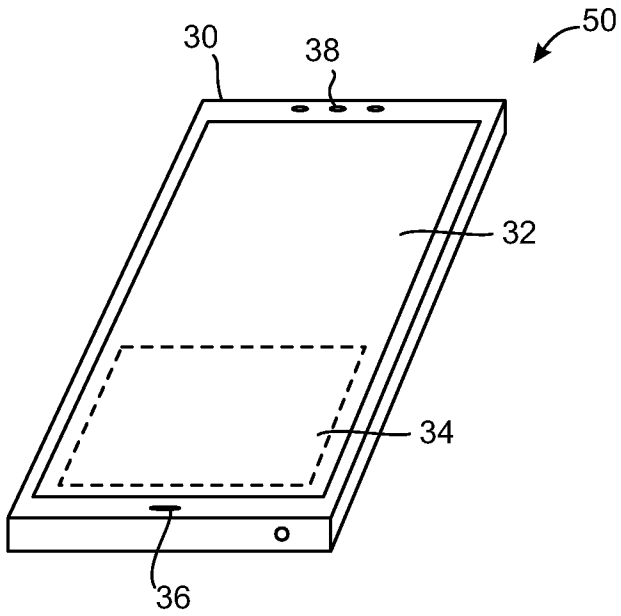
50
30
38
32
34
36

Fig. 2

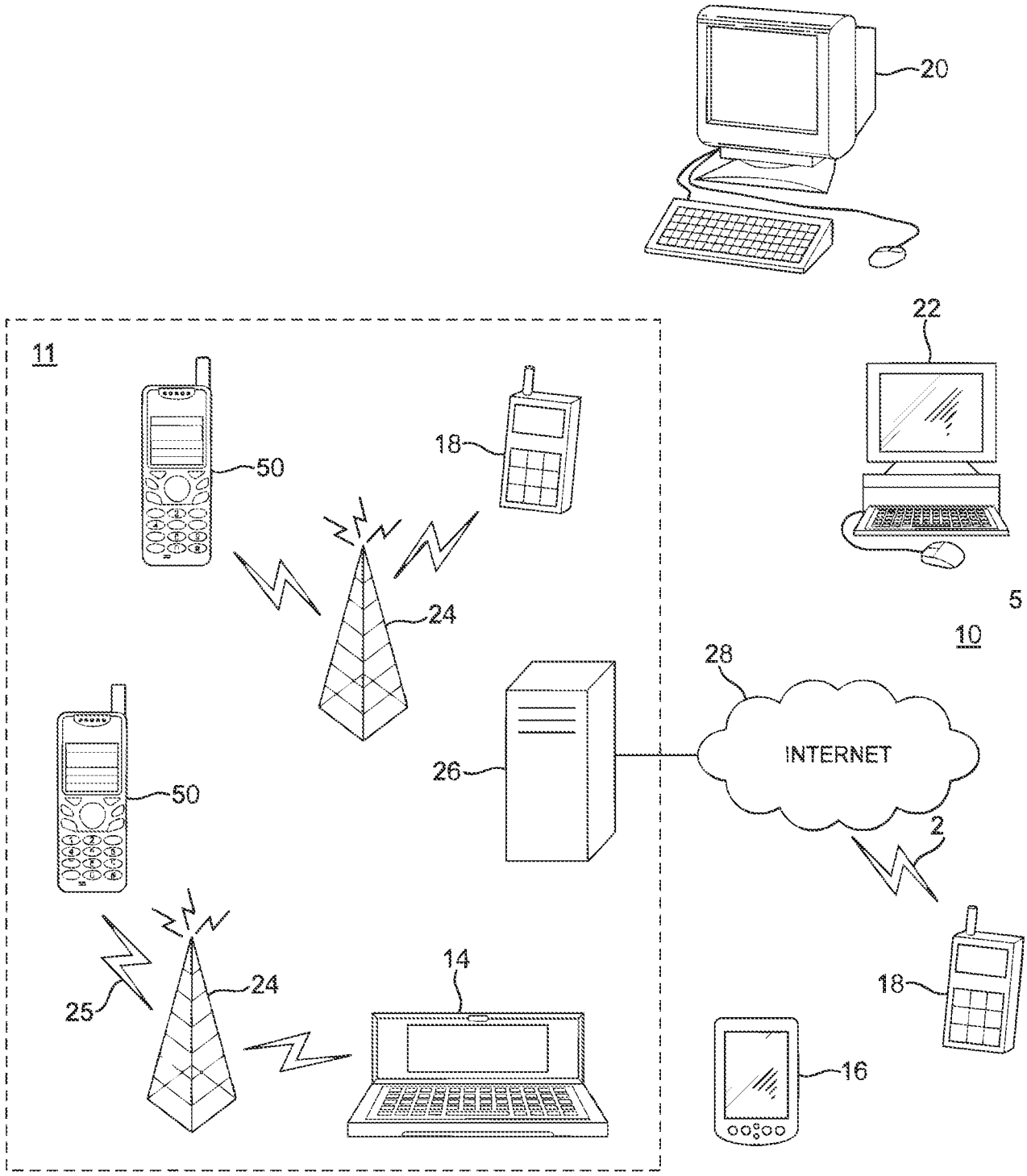

Fig. 3

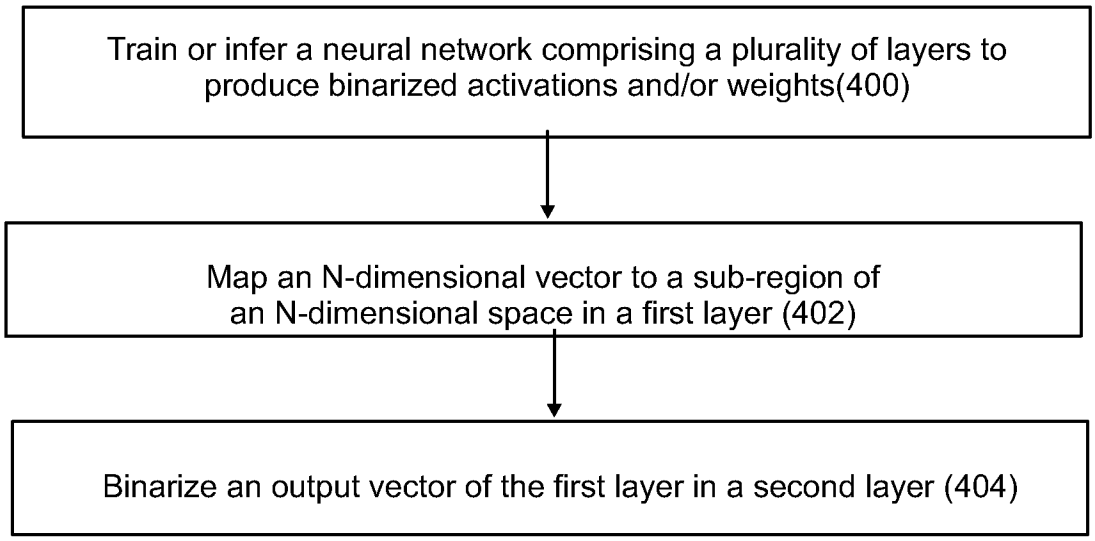

Fig. 4

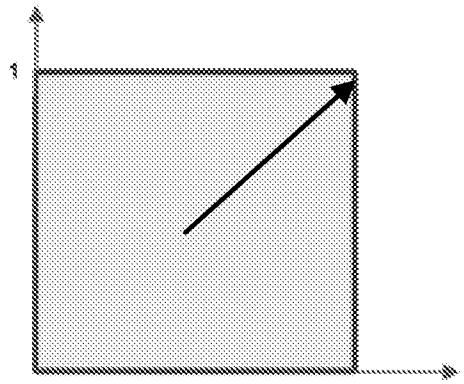

Fig. 5a

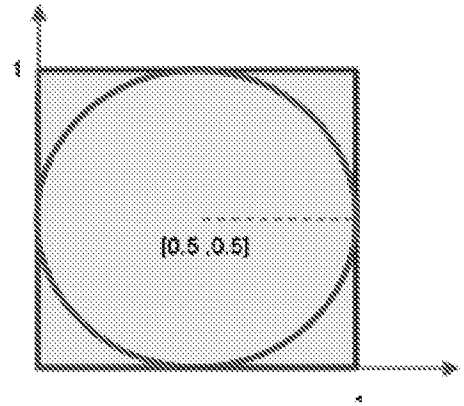

Fig. 5b

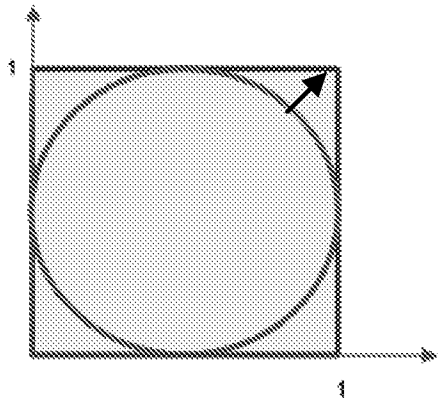

Fig. 5c

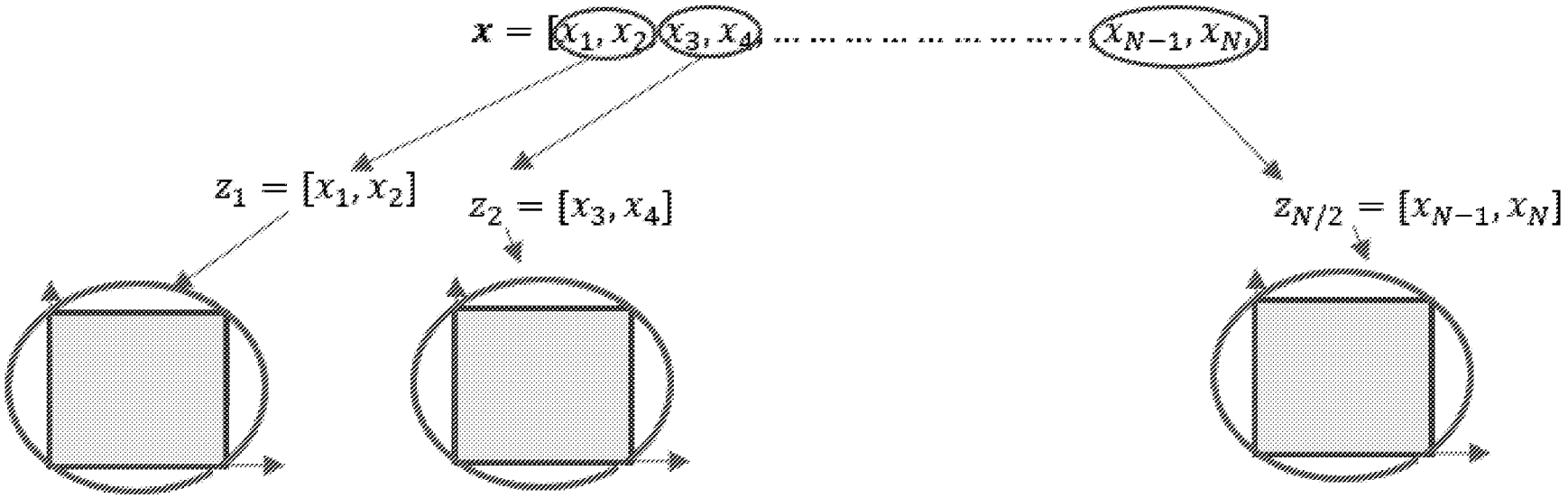

Fig. 6

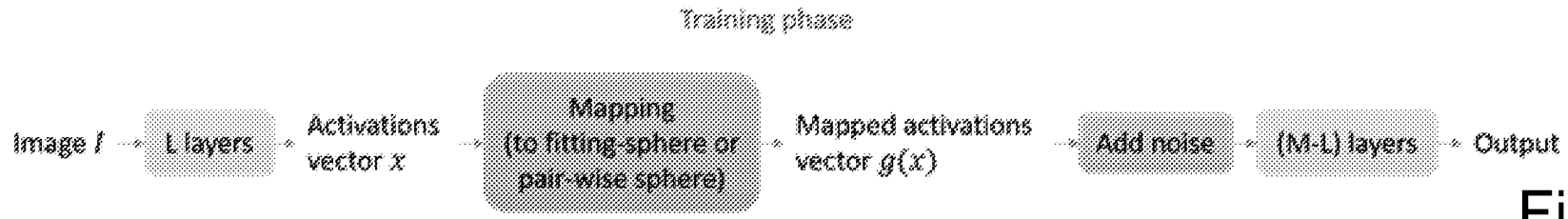

Fig. 7a

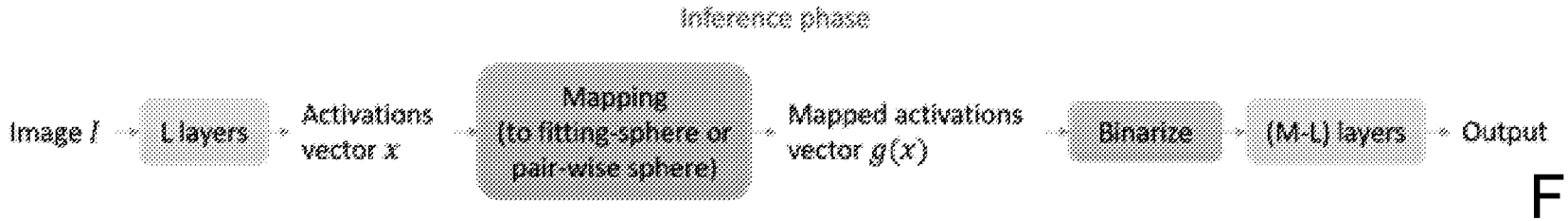

Fig. 7b

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI2019/050495

A. **CLASSIFICATION OF SUBJECT MATTER**

See extra sheet

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)

IPC: G06F, G06N, G06T, H04N

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

FI, SE, NO, DK

Electronic data base consulted during the international search (name of data base, and, where practicable, search terms used)

EPODOC, EPO-Internal full-text databases, Full-text translation databases from Asian languages, WPIAP, XP3GPP, XPAIP, XPESP, XPETSI, XPI3E, XPIEE, XPIETF, XPIOP, XPIPCOM, XPJPEG, XPMISC, XPOAC, XPRD, XPTK, BIOSIS, COMPDX, EMBASE, INSPEC, MEDLINE, TDB, NPL

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MATTHIEU COURBARIAUX et al. Binarized Neural Networks: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1. In: arXiv.org [online], 2016-03-17, [retrieved on 2019-12-04]. Retrieved from <https://arxiv.org/abs/1602.02830>. XP055405835 the whole document, especially see: abstract; sections 1.1, 1.2, 1.6, 3.3, 4; conclusions	1-16
X	CAGLAR AYTEKIN et al. Block-optimized Variable Bit Rate Neural Image Compression. In: arXiv.org [online], 2018-05-28, [retrieved on 2019-12-04]. Retrieved from <https://arxiv.org/abs/1805.10887>. XP080883107 the whole document, especially see: abstract; sections 2.1-2.3, 3, 3.1	1-16

☒ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

*	Special categories of cited documents:	"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
"A"	document defining the general state of the art which is not considered to be of particular relevance	"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
"D"	document cited by the applicant in the international application	"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
"E"	earlier application or patent but published on or after the international filing date	"&"	document member of the same patent family
"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		

Date of the actual completion of the international search	Date of mailing of the international search report
11 December 2019 (11.12.2019)	13 December 2019 (13.12.2019)

Name and mailing address of the ISA/FI
Finnish Patent and Registration Office
FI-00091 PRH, FINLAND
Facsimile No. +358 29 509 5328

Authorized officer
Teppo Häyrynen
Telephone No. +358 29 509 5000

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2018091333 A1 (BOSCH GMBH ROBERT [DE]) 24 May 2018 (24.05.2018) & US 2019266476 A1 (BOSCH GMBH ROBERT [DE]) 29 August 2019 (29.08.2019) paragraphs [0007], [0031]-[0033], [0037]-[0049], [0054]-[0056]; Fig. 1; claims 14, 25	1-16
X	WO 2018048907 A1 (NEOSENSORY INC C/O TMC 260 [US]) 15 March 2018 (15.03.2018) paragraphs [0013]-[0018], [0022]-[0023], [0030], [0038]-[0039], [0060]-[0066]; Figs. 1A, 2A-B, 5	1-16
X	US 2016148078 A1 (SHEN XIAOHUI [US] et al.) 26 May 2016 (26.05.2016) paragraphs [0003]-[0004], [0012]-[0030], [0038]-[0040], [0066]-[0069]; Figs. 2-6	1-16
X	US 2017286830 A1 (EL-YANIV RAN [IL] et al.) 05 October 2017 (05.10.2017) paragraphs [0005], [0008]-[0009], [0014]-[0016], [0020], [0047]-[0055], [0110]; Figs. 1, 2, 6	1-16

WO 2018091333 A1	24/05/2018	CN 109983479 A	05/07/2019
		DE 102016222814 A1	24/05/2018
		EP 3542313 A1	25/09/2019
		US 2019266476 A1	29/08/2019
WO 2018048907 A1	15/03/2018	CN 109688990 A	26/04/2019
		EP 3509549 A1	17/07/2019
		US 2018067558 A1	08/03/2018
		US 10198076 B2	05/02/2019
		US 2019121439 A1	25/04/2019
US 2016148078 A1	26/05/2016	US 9563825 B2	07/02/2017
US 2017286830 A1	05/10/2017	None	

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI2019/050495

CLASSIFICATION OF SUBJECT MATTER

IPC
G06N 3/08 (2006.01)
G06N 3/04 (2006.01)
H04N 19/126 (2014.01)
G06T 9/00 (2006.01)