



(51) International Patent Classification:

G06N 7/00 (2006.01) G06N 20/00 (2019.01)

(21) International Application Number:

PCT/EP2021/085753

(22) International Filing Date:

14 December 2021 (14.12.2021)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

2051484-0 17 December 2020 (17.12.2020) SE

(71) Applicant: ASSA ABLOY AB [SE/SE]; P.O. Box 70340,
107 23 Stockholm (SE).

(72) Inventor: STYLIANIDIS, Matthaïos; Bäckvägen 149,
129 33 Hägersten (SE).

(74) Agent: KRANSELL & WENNBORG KB; P.O. Box
27834, 115 93 Stockholm (SE).

(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, IT, 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, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, 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: ENABLING TRAINING OF AN HMM

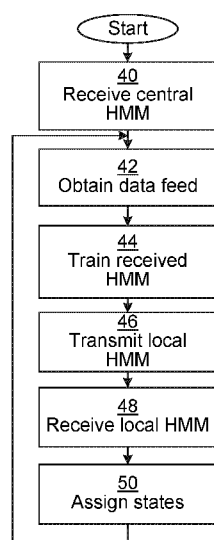


Fig. 2

(57) Abstract: It is provided a method for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of depicting a person. The method is performed by a local training device (1) and comprises: receiving (40) a central HMM from a central node (9); obtaining (42) a data feed capable of depicting the person; training (44) the received HMM based on the data feed, resulting in a locally trained HMM; transmitting (46) the locally trained HMM to the central node (9) for aggregation with other locally trained HMMs; receiving (48) an updated HMM from the central node (9); and assigning (50) each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.



ENABLING TRAINING OF AN HMM

TECHNICAL FIELD

[0001] The present disclosure relates to the field of training a hidden Markov model (HMM) and in particular to enabling training of an HMM for monitoring a person based on a data feed capable of depicting a person.

BACKGROUND

[0002] New technology opens up new opportunities. For instance, the evolution of digital cameras and communication technologies enable monitoring of people to be provided using video surveillance at relatively low cost. This can be particularly useful for elderly people or disabled people, who in this way can enjoy greatly improved quality of life by living in their own home instead of being in a staffed care facility.

[0003] Video surveillance is certainly useful, but privacy issues arise. Hardly anyone enjoys being continuously monitored using video surveillance, for monitoring of when the person needs help.

[0004] One way to reduce the privacy concern is to, instead of manual monitoring, use machine learning models to determine the state of a monitored person. However, machine learning models need to be trained. When video data or other imaging data is input data when the machine learning model is in operation, the machine learning model first needs to be trained, e.g. with video data. Video data may be needed for labelling e.g. even when only 16 x 16 infrared (IR) images are used in operation, since the IR images are not detailed enough for people to see what is happening. Hence video data for training sometimes needs to be manually processed as part of the training process, e.g. for labelling, which is a privacy concern for the person captured in the video data.

SUMMARY

[0005] One object is to enable training that does not compromise privacy of the person for which data is captured for the training data.

[0006] According to a first aspect, it is provided a method for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of

depicting a person. The method is performed by a local training device. The method comprises: receiving a central HMM from a central node; obtaining a data feed capable of depicting the person; training the received HMM based on the data feed, resulting in a locally trained HMM; transmitting the locally trained HMM to the central node for aggregation with other locally trained HMMs; receiving an updated HMM from the central node; and assigning each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.

[0007] The set of monitoring states may contain a first monitoring state indicating presence of at least one person and a second monitoring state indicating absence of people.

[0008] The training the received HMM may comprises adjusting state transition probabilities in the HMM.

[0009] The assigning may be based on a labelled dataset.

[0010] The labelled dataset may comprise a plurality of subsets.

[0011] The subsets of the plurality of subsets may differ in terms of a period of time that is covered.

[0012] Each subset may contain its own label, corresponding to a correct monitored state.

[0013] In one embodiment, the labelled dataset is only used for assigning the states of the updated HMM with monitoring states.

[0014] The data feed may be based on an infrared sensor.

[0015] According to a second aspect, it is provided a local training device for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of depicting a person. The local training device comprises: a processor; and a memory storing instructions that, when executed by the processor, cause the local training device to: receive a central HMM from a central node; obtain a data feed capable of depicting the person; train the received HMM based on the data feed, resulting in a locally trained HMM; transmit the locally trained HMM to the central node for aggregation with other locally trained HMMs; receive an updated HMM

from the central node; and assign each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.

[0016] The set of monitoring states may contain a first monitoring state indicating presence of at least one person and a second monitoring state indicating absence of people.

[0017] The instructions to train the received HMM may comprise instructions that, when executed by the processor, cause the local training device to adjusting state transition probabilities in the HMM.

[0018] The instructions to assign may comprise instructions that, when executed by the processor, cause the local training device to assign based on a labelled dataset.

[0019] The labelled dataset may comprise a plurality of subsets.

[0020] The subsets of the plurality of subsets may differ in terms of a period of time that is covered.

[0021] Each subset may contain its own label, corresponding to a correct monitored state.

[0022] In one embodiment, the labelled dataset is only used for assigning the states of the updated HMM with monitoring states.

[0023] The data feed may be based on an infrared sensor.

[0024] According to a third aspect, it is provided a computer program for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of depicting a person. The computer program comprises computer program code which, when executed on a local training device causes the local training device to: receive a central HMM from a central node; obtain a data feed capable of depicting the person; train the received HMM based on the data feed, resulting in a locally trained HMM; transmit the locally trained HMM to the central node for aggregation with other locally trained HMMs; receive an updated HMM from the central node; and assign each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.

[0025] According to a fourth aspect, it is provided a computer program product comprising a computer program according to the third aspect and a computer readable means on which the computer program is stored.

[0026] Generally, all terms used in the claims are to be interpreted according to their ordinary meaning in the technical field, unless explicitly defined otherwise herein. All references to "a/an/the element, apparatus, component, means, step, etc." are to be interpreted openly as referring to at least one instance of the element, apparatus, component, means, step, etc., unless explicitly stated otherwise. The steps of any method disclosed herein do not have to be performed in the exact order disclosed, unless explicitly stated.

BRIEF DESCRIPTION OF THE DRAWINGS

[0027] Aspects and embodiments are now described, by way of example, with reference to the accompanying drawings, in which:

[0028] Fig 1 is a schematic diagram illustrating an environment in which embodiments presented herein can be applied;

[0029] Fig 2 is a flow chart illustrating embodiments of methods for enabling training of an HMM for monitoring a person based on a data feed capable of depicting a person;

[0030] Fig 3 is a schematic diagram illustrating components of the local training device of Fig 1; and

[0031] Fig 4 shows one example of a computer program product comprising computer readable means.

DETAILED DESCRIPTION

[0032] The aspects of the present disclosure will now be described more fully hereinafter with reference to the accompanying drawings, in which certain embodiments of the invention are shown. These aspects may, however, be embodied in many different forms and should not be construed as limiting; rather, these embodiments are provided by way of example so that this disclosure will be thorough

and complete, and to fully convey the scope of all aspects of invention to those skilled in the art. Like numbers refer to like elements throughout the description.

[0033] According to embodiments presented herein, it is provided training of a Hidden Markov Model (HMM) based on federated learning. Specifically, a central HMM is received in a local training device. The received HMM is trained using local data, and is provided to the central node for aggregation with other locally trained HMMs. The aggregated HMM is then received from the central node in the local node. This achieves unsupervised learning that is aggregated from several local training sessions. Due to the use of unsupervised training, the local and central (aggregated) HMMs can be trained with vast amounts of unlabelled data feeds from separate local HMMs. Since the data feeds do not need to leave the local site where monitoring occurs, neither for training nor for inference, and the aggregation is done in a secure way, privacy concerns of the monitored person are addressed. The resulting HMM can predict, based on a data feed of a number of frames, the probability of a certain monitored state at a certain point in time in the future.

[0034] Fig 1 is a schematic diagram illustrating an environment in which embodiments presented herein can be applied. A person 5 to be monitored is at least part of the time present in a physical space 14. The physical space 14 can e.g. be a room, a flat, a house, an office etc. A monitoring device 2 is configured to monitor the person 5 based on a sensor device 3 forming part of the monitoring device 2 or locally connected to the monitoring device 2. The monitoring device 2 is also used for capturing data of the person 5 for purposes of training a local HMM 4 and a central HMM 9. The sensor device 3 provides a data feed capable of depicting the person 5, e.g. as a sequence of images (i.e. a video sequence). The sensor device 3 can be implemented as an infrared (IR) camera, a video camera, a lidar, a radar or any other suitable imaging technology. Additional sensor devices (not shown) can also be provided, providing respective data feeds. A local training device 1 is used for training the local HMM 4 and for contributing to the training of the central HMM 9 as described in more detail below. The local training device 1 can be provided as part of the monitoring device 2 or separately. In any case, the local training device 1 is provided at the same site as the sensor device 3. In this way, sensitive data feeds from the sensor device 3 can be used by the local training device 1 without the data feed having to be communicated remotely.

[0035] The monitoring device 2 comprises the local HMM 4. There may be one or more monitoring devices 2 working in parallel on the same or complementing scene. The monitoring device 2 can be connected to a network 6, which can be an Internet protocol (IP) based network. The network 6 can e.g. comprise any one or more of a local wireless network, a cellular network, a wired local area network, a wide area network (such as the Internet), etc. A central node 7, containing the central HMM 9, is also connected to the network 6.

[0036] The central node 7 can then be used in cooperation with a plurality of separate monitoring devices 2 with respective local HMMs 4 and local training devices 1, as part of a federated ML (machine learning) model.

[0037] The local HMM 4 of the monitoring device 2 is used to predict current or future monitored states based on the data feed from the sensor device 3. Specifically, the local HMM 4 is used to infer monitoring states of the person 5. Non-limiting examples of monitoring states, all relating to the person, are: absent, present, lying in bed, lying on floor, breathing, distress, etc. The probability of transitioning from one state to another is denoted a state transition probability.

[0038] The HMMs used herein can e.g. be Gaussian HMMs. In the HMM, there are a number of states. However, the states are hidden can only be observed indirectly, via observations. The observations can be expressed as an observation vector y .

[0039] For the purposes of monitoring people in the examples presented herein, if there is an IR image of 16x16 pixels as an input, this is represented in an observation vector y of $16 * 16$ elements. The observations in this vector have values in every time step, comprising the IR image.

[0040] Each state can have a mean vector and covariance matrix parameters. In the Gaussian HMM, there is an assumption that the observation vector is generated in a Gaussian distribution. In this way, given the parameters of each state, the probability of the vector y (IR image) can be estimated to have a certain continuous value X .

[0041] In essence, an observed vector y results in a value X at a certain timestep.

[0042] It is to be noted that the HMM can be made up of any suitable number of states and observation vector size.

[0043] Fig 2 is a flow chart illustrating embodiments of methods for enabling training of an HMM for monitoring a person based on a data feed capable of depicting a person. The embodiments are performed by the local training device 1.

[0044] In a receive central HMM step 40, the local training device 1 receives a central HMM from a central node 9, over the network 6.

[0045] In an *obtain data feed* step 42, the local training device 1 obtains a data feed capable of depicting the person. The data feed can be based on an infrared sensor, which has the advantage of being able to depict people regardless of lighting conditions. Alternatively or additionally, the data feed is based on other sensors, as long as the sensor is capable of depicting the person. The data feed can be pre-processed as part of this step, e.g. for consistent aligning of the image, etc., as known in the art per se.

[0046] In a *train received HMM* step 44, the local training device 1 trains the received (now local) HMM based on the data feed, resulting in a locally trained HMM. The training can comprise adjusting state transition probabilities in the HMM. The training can e.g. occur using a Baum–Welch algorithm, allowing the training of the HMM without any labelling, or Viterbi training.

[0047] In a *transmit local HMM* step 46, the local training device 1 transmits the locally trained HMM to the central node 9 for aggregation with other locally trained HMMs. The aggregation of the locally trained HMMs can be based on accumulating or averaging the state transition probabilities from the different locally trained HMMs, resulting in a central HMM. Another example of how aggregation can occur is described in Yurochkin et al. “Statistical Model Aggregation via Parameter Matching”, 33rd Conference on Neural Information Processing Systems (NEURIPS 2019), Vancouver, Canada. It is to be noted that the data feed itself never leaves the local site and is not needed for the central aggregation of local HMMs.

[0048] The locally trained HMM can be defined as the currently defined state transition probabilities.

[0049] In a *receive updated HMM* step 48, the local training device 1 receives an updated HMM from the central node 9. The updated HMM is the aggregate central HMM based on the locally trained HMMs. In this way, the updated HMM benefits from

training based on multiple data feeds in multiple different local training devices at different sites, even though the data feeds never need to leave the local sites, thus preserving privacy for people at the different sites.

[0050] In an *assign states* step 50, the local training device 1 assigns each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person. For instance, the set of monitoring states can contain a first monitoring state indicating presence of at least one person and a second monitoring state indicating absence of people. This assignment step is performed since the HMM is incapable in itself of determining which of its hidden states corresponds to a monitored state. The assigning can be based on a labelled dataset, where each dataset comprises a plurality of subsets. Each subset covers a period of time of the data feed, e.g. containing a number of images. Each labelled dataset defines, in its labelling its associated correct monitored state. In example for presence/absence detection, if our dataset comprises N subsets, each contains M images, we would have a series of N labels, one for each subset. If each subset is composed of 20 consecutive images, the label would indicate whether there is a person present or not, e.g. at the time corresponding to the last image in each subset.

[0051] The local HMM determines a (HMM) state based on the labelled dataset and, using the label, the monitored state can be assigned to the HMM state. The labelled dataset is only used for assigning the states of the HMM with monitoring states and not for training the local HMM. Hence the labelled dataset does not need to be extensive and can be quite small. The labelled dataset only needs to be distinct enough to enable the labelling of each HMM state to a monitored state. It is to be noted that the *assign states* step 50 might not be needed to be performed for each iteration of the method, as long as it is performed at least once.

[0052] Using embodiments presented herein, due to the use of unsupervised training, the local and central (aggregated) HMMs can be trained with vast amounts of unlabelled data feeds from separate local HMMs. Since the data feeds do not need to leave the site of the monitoring device (or even the monitoring device at all, when the local training device forms part of the monitoring device) and the aggregation is done in a secure way, privacy concerns of the monitored person are addressed. The resulting

HMM can predict, based on a data feed of a number of frames, the probability of a certain monitored state at a certain point in time in the future.

[0053] Fig 3 is a schematic diagram illustrating components of the local training device 1 of Fig 1. It is to be noted that, when the local training device 1 is implemented in a host device such as the monitoring device 2, one or more of the mentioned components can be shared with the host device. A processor 60 is provided using any combination of one or more of a suitable central processing unit (CPU), multiprocessor, microcontroller, digital signal processor (DSP), etc., capable of executing software instructions 67 stored in a memory 64, which can thus be a computer program product. The processor 60 could alternatively be implemented using an application specific integrated circuit (ASIC), field programmable gate array (FPGA), etc. The processor 60 can be configured to execute the method described with reference to Fig 2 above.

[0054] The memory 64 can be any combination of random-access memory (RAM) and/or read-only memory (ROM). The memory 64 also comprises persistent storage, which, for example, can be any single one or combination of magnetic memory, optical memory, solid-state memory or even remotely mounted memory.

[0055] A data memory 66 is also provided for reading and/or storing data during execution of software instructions in the processor 60. The data memory 66 can be any combination of RAM and/or ROM.

[0056] The local training device 1 further comprises an I/O interface 62 for communicating with external and/or internal entities. For instance, the I/O interface 62 allows the local training device 1 to communicate the network 6. Optionally, the I/O interface 62 also includes a user interface.

[0057] Other components of the local training device 1 are omitted in order not to obscure the concepts presented herein.

[0058] Fig 4 shows one example of a computer program product 90 comprising computer readable means. On this computer readable means, a computer program 91 can be stored, which computer program can cause a processor to execute a method according to embodiments described herein. In this example, the computer program product is in the form of a removable solid-state memory, e.g. a Universal Serial Bus

(USB) drive. As explained above, the computer program product could also be embodied in a memory of a device, such as the computer program product 64 of Fig 3. While the computer program 91 is here schematically shown as a section of the removable solid-state memory, the computer program can be stored in any way which is suitable for the computer program product, such as another type of removable solid-state memory, or an optical disc, such as a CD (compact disc), a DVD (digital versatile disc) or a Blu-Ray disc.

[0059] The aspects of the present disclosure have mainly been described above with reference to a few embodiments. However, as is readily appreciated by a person skilled in the art, other embodiments than the ones disclosed above are equally possible within the scope of the invention, as defined by the appended patent claims. Thus, while various aspects and embodiments have been disclosed herein, other aspects and embodiments will be apparent to those skilled in the art. The various aspects and embodiments disclosed herein are for purposes of illustration and are not intended to be limiting, with the true scope and spirit being indicated by the following claims.

CLAIMS

1. A method for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of depicting a person, the method being performed by a local training device (1), the method comprising:
 - receiving (40) a central HMM from a central node (9);
 - obtaining (42) a data feed capable of depicting the person;
 - training (44) the received HMM based on the data feed, resulting in a locally trained HMM;
 - transmitting (46) the locally trained HMM to the central node (9) for aggregation with other locally trained HMMs;
 - receiving (48) an updated HMM from the central node (9); and
 - assigning (50) each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.
2. The method according to claim 1, wherein the set of monitoring states contain a first monitoring state indicating presence of at least one person and a second monitoring state indicating absence of people.
3. The method according to any one of the preceding claims, wherein the training (44) the received HMM comprises adjusting state transition probabilities in the HMM.
4. The method according to any one of the preceding claims, wherein the assigning (50) is based on a labelled dataset.
5. The method according to claim 4, wherein the labelled dataset comprises a plurality of subsets.
6. The method according to claim 5, wherein the subsets of the plurality of subsets differ in terms of a period of time that is covered.
7. The method according to any claim 5 or 6, wherein each subset contains its own label, corresponding to a correct monitored state.
8. The method according to any one of claims 4 to 7, wherein the labelled dataset is only used for assigning the states of the updated HMM with monitoring states.
9. The method according to any one of the preceding claims, wherein the data feed is based on an infrared sensor.

10. A local training device (1) for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of depicting a person, the local training device (1) comprising:

a processor (60); and

a memory (64) storing instructions (67) that, when executed by the processor, cause the local training device (1) to:

receive a central HMM from a central node (9);

obtain a data feed capable of depicting the person;

train the received HMM based on the data feed, resulting in a locally trained HMM;

transmit the locally trained HMM to the central node (9) for aggregation with other locally trained HMMs;

receive an updated HMM from the central node (9); and

assign each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.

11. The local training device (1) according to claim 10, wherein the set of monitoring states contain a first monitoring state indicating presence of at least one person and a second monitoring state indicating absence of people.

12. The local training device (1) according to any one of claims 10 to 11, wherein the instructions to train the received HMM comprise instructions (67) that, when executed by the processor, cause the local training device (1) to adjusting state transition probabilities in the HMM.

13. The local training device (1) according to any one of claims 10 to 12, wherein the instructions to assign instructions (67) that, when executed by the processor, cause the local training device (1) to assign based on a labelled dataset.

14. The local training device (1) according to claim 13, wherein the labelled dataset comprises a plurality of subsets.

15. The local training device (1) according to claim 14, wherein the subsets of the plurality of subsets differ in terms of a period of time that is covered.

16. The local training device (1) according to claim 14 or 15, wherein each subset contains its own label, corresponding to a correct monitored state.

17. The local training device (1) according to any one of claims 13 to 16, wherein the labelled dataset is only used for assigning the states of the updated HMM with monitoring states.
18. The local training device (1) according to any one of claims 10 to 17, wherein the data feed is based on an infrared sensor.
19. A computer program (67, 91) for enabling training of a hidden Markov model, HMM, for monitoring a person based on a data feed capable of depicting a person, the computer program comprising computer program code which, when executed on a local training device (1) causes the local training device (1) to:
- receive a central HMM from a central node (9);
 - obtain a data feed capable of depicting the person;
 - train the received HMM based on the data feed, resulting in a locally trained HMM;
 - transmit the locally trained HMM to the central node (9) for aggregation with other locally trained HMMs;
 - receive an updated HMM from the central node (9); and
 - assign each state in the updated HMM to one state of a set of monitoring states, wherein each monitoring state indicates a state of the person.
20. A computer program product (64, 90) comprising a computer program according to claim 19 and a computer readable means on which the computer program is stored.

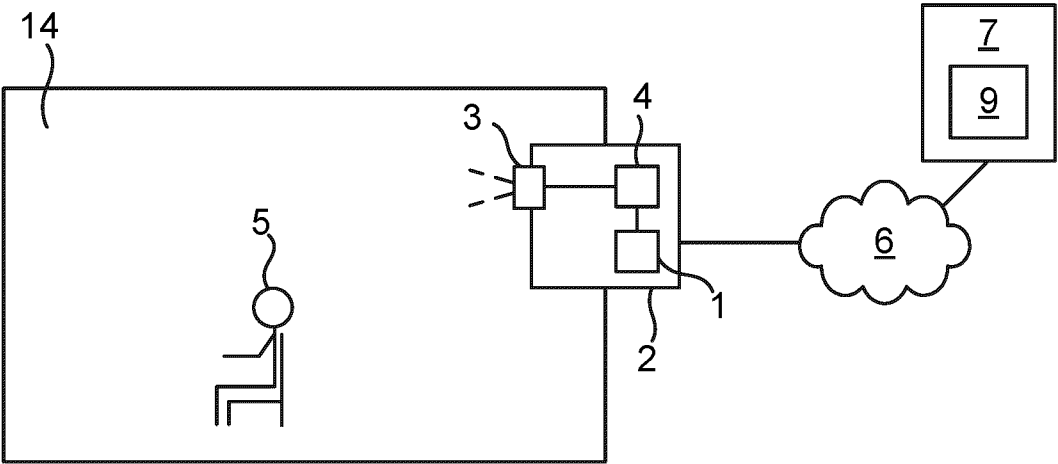


Fig. 1

2/2

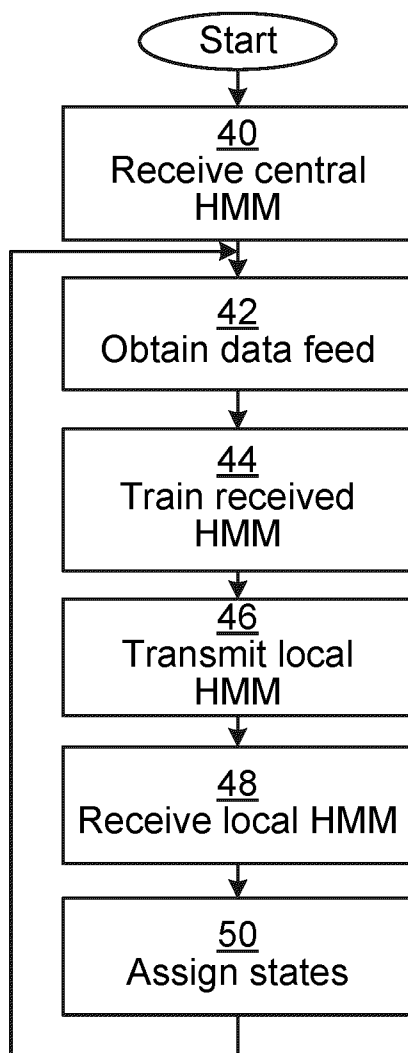


Fig. 2

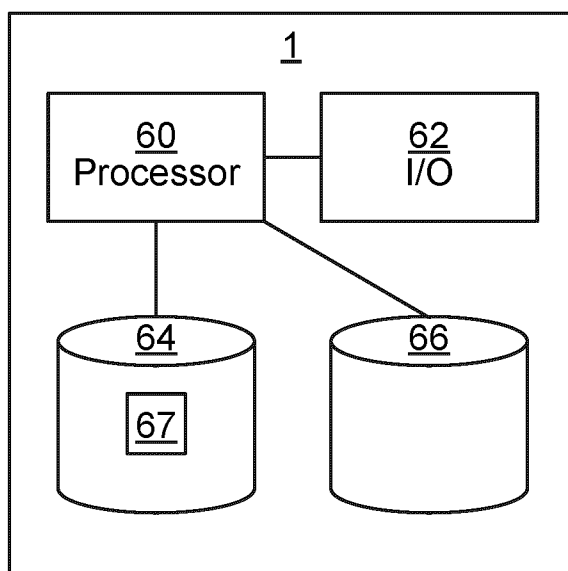


Fig. 3

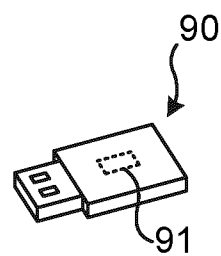


Fig. 4

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2021/085753

A. CLASSIFICATION OF SUBJECT MATTER INV. G06N7/00 G06N20/00 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		
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.
X	MIKHAIL YUROCHKIN ET AL: "Statistical Model Aggregation via Parameter Matching", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 1 November 2019 (2019-11-01), XP081527519, cited in the application abstract page 1, line 1 - page 9, last line page 12, line 1 - page 16, last line ----- -/--	1-20
☒ 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
8 April 2022		26/04/2022
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Totir, Felix

INTERNATIONAL SEARCH REPORT

International application No

PCT/EP2021/085753

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	SASSI IMAD ET AL: "A New Improved Baum-Welch Algorithm for Unsupervised Learning for Continuous-Time HMM Using Spark", INTERNATIONAL JOURNAL OF INTELLIGENT ENGINEERING AND SYSTEMS, vol. 13, no. 1, 29 February 2020 (2020-02-29), pages 214-226, XP055909182, DOI: 10.22266/ijies2020.0229.20 abstract page 214, left-hand column, line 1 - page 224, right-hand column, line 20 -----	1-20
A	Anikeev Maxim ET AL: "Parallel Implementation of Baum-Welch Algorithm AutoManSec 4 CloudIoT -Autonomic Management and Security for Cloud and IoT View project Parallel Implementation of Baum-Welch Algorithm", / 1 January 2006 (2006-01-01), pages 1-4, XP055909160, Retrieved from the Internet: URL:https://www.researchgate.net/publication/260205306_Parallel_Implementation_of_Baum-Welch_Algorithm [retrieved on 2022-04-05] abstract page 1, left-hand column, line 1 - page 3, right-hand column, last line -----	1-20