



(51) International Patent Classification:

G06N 3/084 (2023.01) G06T 7/00 (2017.01)
G06N 3/0464 (2023.01) G16H 30/40 (2018.01)
G06N 20/00 (2019.01) G16H 50/20 (2018.01)

(21) International Application Number:

PCT/AU2024/051378

(22) International Filing Date:

19 December 2024 (19.12.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

2023904194 22 December 2023 (22.12.2023) AU
2024903443 23 October 2024 (23.10.2024) AU

(71) Applicant: **COMMONWEALTH SCIENTIFIC AND
INDUSTRIAL RESEARCH ORGANISATION**
[AU/AU]; Clunies Ross St, Acton, Queensland 2601 (AU).

(72) Inventors: **MEJAN-FRIPP, Jurgen**; c/- Clunies Ross St, Acton, Australian Capital Territory 2601 (AU). **BOURGEAT, Pierrick**; c/- Clunies Ross St, Acton, Australian Capital Territory 2601 (AU). **DORE, Vincent**; c/- Clunies Ross St, Acton, Australian Capital Territory 2601 (AU). **LEBRAT, Leo**; c/- Clunies Ross St, Acton, Australian Capital Territory 2601 (AU).

(74) Agent: **FB RICE PTY LTD**; Level 11, 1 Eagle Street, Brisbane, Queensland 4000 (AU).

(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, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO,

(54) Title: TRAINING DEEP LEARNING MODELS

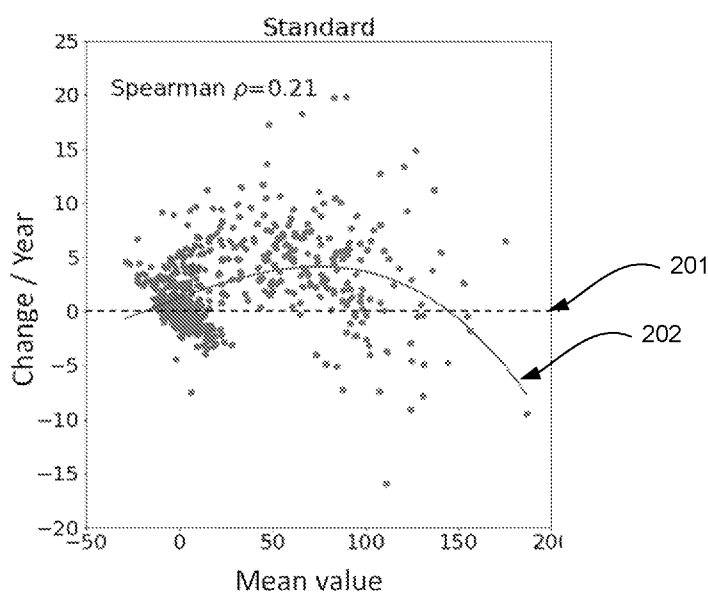


Fig. 2

(57) Abstract: This disclosure relates to training a machine learning model on longitudinal data. More particularly, this disclosure relates to training a machine learning model on longitudinal medical scan images of a patient study. A processor calculates two or more predictions of a biomedical factor by performing for each of two or more of the medical images: applying the machine learning model to the medical image to calculate an output; and calculating a prediction using the output of the machine learning model for the medical image. The processor then calculates a rate of change between the two or more predictions for the two or more medical images. The processor then creates a trained machine learning model by updating weights of the machine learning model to minimise a loss value, wherein the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change.

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, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, 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, ME, 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))*
- *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

"Training deep learning models"

Cross-Reference to Related Applications

[0001] The present application claims priority from Australian Provisional Patent Application No 2023904194 filed on 22 December 2023 and Australian Provisional Patent Application No 2024903443 filed on 23 October 2024, the contents of which are incorporated herein by reference in their entirety.

Technical Field

[0002] This disclosure relates to training a machine learning model on longitudinal data. More particularly but not necessarily exclusively, this disclosure relates to training a machine learning model on longitudinal medical scan images of a patient study.

Background

[0003] Machine learning models, such as deep neural networks, are powerful tools in addressing challenging problems and providing predictions. For example, machine learning models can be used to determine the presence of a disease from an image or a collection of images, such as medical images. Such machine learning models are trained and can then be used to make predictions. Training involves comparing the output of the model with ground truth information (i.e., information that is known to be real or true, and is the target output of the model) and updating the machine learning model based on the comparison.

[0004] Machine learning models, particularly models that utilise deep learning techniques, are trained to minimise a loss value (or a loss function) based on one or a combination of metrics based on some ground truth measurements. The loss value is generally calculated from the difference between the output and the ground truth, with the goal of training being to minimise this loss such that the machine learning model produces an output that is near identical to the ground truth. In one example, for models that are used for image segmentation, the loss value is indicative of the overlap between the predicted mask from the model and a manually defined mask of the structure to be segmented in an image. In this

example, the loss value would be formulated in a way that the overlap is maximised by minimising the loss value. Other examples of loss values include the mean square error (MSE) measure between the predicted volume structure and its ground truth volume.

[0005] However, ground truth can often have inherent noise due to many reasons, such as the equipment used to measure the data or other errors. This can make it difficult to train a machine learning model on such ground truth data. In particular, when the training data is longitudinal data, the data is often measured years apart with different machines and calibration settings for each measurement. This introduces noise as there is inconsistency between each successive measurement. Moreover, the noise makes it difficult to train the model on ground truth data that comes from multiple sources. This is often the case when training machine learning models on medical images, where many databases of training data are collated as a significantly large amount of training data is required to accurately train a model.

[0006] Any discussion of documents, acts, materials, devices, articles or the like which has been included in the present specification is not to be taken as an admission that any or all of these matters form part of the prior art base or were common general knowledge in the field relevant to the present disclosure as it existed before the priority date of each of the appended claims.

[0007] Throughout this specification the word “comprise”, or variations such as “comprises” or “comprising”, will be understood to imply the inclusion of a stated element, integer or step, or group of elements, integers or steps, but not the exclusion of any other element, integer or step, or group of elements, integers or steps.

Summary

[0008] A method for training a machine learning model on longitudinal medical scan images of a patient study comprises:

calculating two or more predictions of a biomedical factor by performing for each of two or more of the medical images:

applying the machine learning model to the medical image to calculate an output ; and

calculating a prediction using the output of the machine learning model for the medical image;

calculating a rate of change between the two or more predictions for the two or more medical images; and

creating a trained machine learning model by updating weights of the machine learning model to minimise a loss value, wherein the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change.

[0009] It is an advantage that the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change as the expected rate of change can be enforced into the machine learning model during training. More specifically, training of the machine learning model leverages information that is not available during inference. This is ideal for situations with noisy ground truth data. Moreover, inference of the trained machine learning model can be performed by applying the model to a single image.

[0010] In some embodiments, the method further comprises determining a polynomial function of rate of change of the biomedical factor and calculating the expected rate of change using the polynomial function.

[0011] In some embodiments, calculating the expected rate of change using the polynomial function comprises applying the polynomial function to a mean value of the biomedical factor calculated by an average of the two or more predictions.

[0012] In some embodiments, determining the polynomial function comprises calculating a rate of change and a mean value for each pair of medical images in the patient study.

[0013] In some embodiments, the loss value further comprises a change penalty value reflecting a difference between the two or more predictions, the change penalty value penalising an unexpected temporal trend of the biomedical factor related to the expected rate of change.

[0014] In some embodiments, the loss value further comprises a correction penalty value reflecting a difference between the two or more predictions and two or more corresponding ground truth values, the correction penalty value penalising the two or more predictions from deviating from the two or more corresponding ground truth values.

[0015] In some embodiments, the loss value further comprises a correction penalty value reflecting a difference between the two or more outputs and an over-correction factor, the correction penalty value penalising the two or more outputs from deviating from the over-correction factor.

[0016] In some embodiments, the loss value further comprises an additional penalty value calculated by one or more of a difference between each of the two or more outputs of the machine learning model and a respective ground truth measurement; and a mean square error between each of the two or more outputs and the respective ground truth measurement.

[0017] In some embodiments, the loss value is calculated by a weighted sum of the penalty value, the change penalty value, the correction penalty value, and the additional penalty value.

[0018] In some embodiments, each of the medical images of the patient study has a timestamp and at least two of the two or more medical image have non-adjacent timestamps.

[0019] In some embodiments, the method further comprises calculating a further two or more predictions of the biomedical factor by performing for each of a further two or more medical images of the patient study: applying the trained machine learning model to the medical image to calculate an output; and calculating a prediction using the output; calculating a further rate of change between the further two or more predictions; and creating a further trained machine learning model by updating weights of the trained machine learning model to minimise a further loss value based of the further rate of change.

[0020] In some embodiments, the method further comprises calculating another two or more predictions of the biomedical factor by performing for each of two or more medical images of another patient study different to the patient study: applying the trained machine learning model or further trained machine learning model to the medical image to calculate an output;

and calculating a prediction using the output; calculating another rate of change between the other two or more predictions; and creating another trained machine learning model by updating weights of the trained machine learning model or further trained machine learning model to minimise another loss value based on the other rate of change.

[0021] In some embodiments, the method further comprises applying the trained machine learning model to a medical image of a patient to aid in diagnosis of the patient based on the output of the trained machine learning model.

[0022] In some embodiments, each of the medical images is one of a magnetic resonance image; positron emission tomography (PET) image; or computed tomography (CT) image.

[0023] In some embodiments, the biomedical factor is one of: medical image quantification factor; a biological structure value; or a biomarker.

[0024] In some embodiments, the medical image quantification factor is a Centiloid or standardised uptake value ratio scaling factor and the biological structure value is volume of a biological structure visible in the medical image.

[0025] In some embodiments, each of the medical images is a PET image and the method further comprises performing PET quantification by applying the trained machine learning model to a PET image of a patient.

[0026] In some embodiments, the medical image quantification factor is a Centiloid or standardised uptake value ratio scaling factor; and the method further comprises applying the trained machine learning model to a medical image of a patient to aid in diagnosis of Alzheimer's disease based on the Centiloid or standardised uptake value ratio scaling factor that is an output of the trained machine learning model.

[0027] In some embodiments, the method further comprises generating one or more masks that correspond to corrections determined by the trained machine learning model.

[0028] In some embodiments, the machine learning model is a neural network comprising one or more convolutional layers.

[0029] Software, when installed on a computer and executed by the computer, causes the computer to perform the above method.

[0030] A system for training a machine learning model on longitudinal medical scan images of a patient study comprises:

a processor configured to:

calculate two or more predictions of a biomedical factor by performing for each of two or more of the medical images:

applying the machine learning model to the medical image to calculate an output; and

calculating a prediction using the output of the machine learning model for the medical image;

calculate a rate of change between the two or more predictions for the two or more medical images; and

create a trained machine learning model by updating weights of the machine learning model to minimise a loss value, wherein the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change.

[0031] Optional features provided in relation to the method, equally apply as optional features to the software and the system.

Brief Description of Drawings

[0032] An example will be described with reference to the following drawings:

[0033] Fig. 1 shows a plot of calculated Centiloid values over time calculated using longitudinal PET scan images.

[0034] Fig. 2 shows a plot of the rate of change of the Centiloid value verses the mean value of the Centiloid value.

[0035] Fig. 3 illustrates a system for training a machine learning model on longitudinal medical scan images of a patient study.

[0036] Fig. 4 illustrates a method for training a machine learning model on longitudinal medical scan images of a patient study.

[0037] Fig. 5 is an example machine learning model with a convolutional neural network (CNN) architecture.

[0038] Fig. 6 shows the progression of the calculated rate of change of the scaled Centiloid values as the training of the machine learning model increases, in the example embodiment.

[0039] Fig. 7 shows the progression of the Standardized Uptake Value Ratio (SUVR) scaling factors as the optimisation processes described herein continues.

[0040] Fig. 8a shows initialised target masks used to generate optimised target masks.

[0041] Fig. 8b shows the optimised target masks with similar SUVR scaling factors to the SUVR provided by the trained machine learning model disclosed herein.

[0042] Fig. 9a shows initialised reference masks used to generate optimised reference masks.

[0043] Fig. 9b shows the optimised reference masks with similar SUVR scaling factors to the SUVR provided by the trained machine learning model disclosed herein.

Description of Embodiments

[0044] The disclosed system and method relate to training a machine learning model on longitudinal data to address the problem of noisy ground truth training data. The disclosed system and method utilise the inherent or expected properties of the longitudinal data as a source of additional information to reduce the effects of noisy ground truth data. More specifically, longitudinal data that exhibits an expected rate of change of a factor, such as a

biomedical factor, can be used to train a machine learning model by formulating the training in a way that reflects this expected rate of change.

[0045] Longitudinal data includes successive measurements of the same subject over a period of time. For example, longitudinal data may be a positron emission tomography (PET) image or magnetic resonance (MR) image of the brain repeated on the same participant every year for N number of years. This is commonly performed on patients who are at risk of Alzheimer's disease to determine the progression of the disease by studying the amount of amyloid plaques in the brain over time. As Alzheimer's disease is a progressive disorder, it is expected that the amount of amyloid plaques in the brain increases over time, which is an expected trend observed from the longitudinal data of a patient with this disease.

[0046] To utilise the inherent or expected properties for training a machine learning model, the disclosed system and method uses a class of penalty values (as referred to as longitudinal constraints) in the loss value. More specifically, the penalty values exploit the expected rate of change of the longitudinal data by penalising the unexpected rate of change. A simple longitudinal constraint could be to penalise negative (positive) changes between baseline and follow-up when the change is expected to be positive (negative). These changes could be biomedical factors such as the PET quantification within a mask or the volume of a segmentation mask, or other (more involved) factors. For example, in relation of Centiloid values, as the expected trend is that the rate of change is positive (i.e., the Centiloid values increase over time), a penalty value can be formulated to penalise training the machine learning model on negative rates of change.

[0047] This disclosure also provides a framework to include these constraints on a single timepoint inference model. In other words, while the model is trained to use the longitudinal information inherent to the longitudinal data for training (e.g., multiple images), the trained model does not require the longitudinal information during inference. This means that the trained model can be used to calculate predictions on a single timepoint during inference (e.g., a single image). One of the main advantages of the disclosed system and method is that it enables the inclusion of longitudinal constraints without relying on multiple timepoints during inference.

[0048] The proposed constraints or penalty values can also complement existing losses by ensuring longitudinal consistency in the training set, instead of solely relying on a noisy “ground truth” data. When properly trained, this also translates into better longitudinal consistency in the testing set, even though each image is inferred (tested) independently from all other timepoints. In particular, as longitudinal data can be captured over many years using machines with different calibration setting each time or completely different machines all together, or in the case of PET imaging, using different PET tracers, the proposed penalty values enable a normalisation of the longitudinal data, which in turn enables machine learning models to be trained on such noisy data.

[0049] For example, the overall trajectory of expected change for a mean measure can be computed from a large population analysis. This overall trajectory can use the error between the actual change and expected change as an additional loss. Since any information can be added in the loss, additional information can also be included into the loss, such as demographics. In another example, where different trajectories are expected for different clinical diagnosis, these different trajectories can be used to define different losses for different clinical groups. Since this information is not used at inference time (in other words, it is only used to compute the loss during training), this limits the risk to overfit the model to the data.

[0050] The disclosed system and method are applicable to many different machine learning (specifically, deep learning) applications, where longitudinal data is available, and where there are known longitudinal constraints that can be applied based on expected trends in the longitudinal data. For example, the machine learning application may be image segmentation, where there is a model of change over time of expectations with respect to increase/decrease of volume of structure over time. In another example, when training a model to predict the volume of a structure that is expected to shrink over time, this information can be used in the loss function in combination with other standard losses, and in this case, penalise the volume increasing over time. This is particularly useful for problems where the ground truth might not be very reliable. For example, this is particularly useful for PET quantification, since PET images can be very noisy.

[0051] In particular, PET quantification is used to analyse PET images and extract quantitative information from these images. PET quantification for the study of Alzheimer's disease involves a normalization of the PET images to the Centiloid scale, which is described as the standard scale for PET images acquired using an amyloid PET tracer. In a general sense, PET images are typically normalised using the Standardized Uptake Value Ratio or SUVR. The SUVR is defined as the ratio of a region of the PET image containing specific binding (a target region where the PET tracer will bind to the target), and a region of the PET image containing non-specific binding (as referred to as a reference region). In the case of amyloid imaging, those are defined as a region containing amyloid, typically the neocortex, and a region containing no amyloid, typically the cerebellum. Since there exists different PET tracers that bind to amyloid, each having different pharmacokinetic properties, the Centiloid scale was developed to harmonise the quantifications (SUVR) from all amyloid PET tracers into the same scale using a linear transform based on head-to-head paired data. In other words, while the SUVR is typically used for most PET tracer quantification, in the specific case of Amyloid PET imaging, the Centiloid scale is typically used as it allows to quantify different PET tracers using the same scale. Due to the progressive nature of Alzheimer's disease, one would expect that the unit of the Centiloid scale increases over time as the amount of amyloid deposits increases in the brain. However, by plotting the calculated Centiloid values over time, it can be shown that this is not always the observed trend in the data.

[0052] Fig. 1 shows a plot of calculated Centiloid values over time calculated using longitudinal PET scan images. The slope of the lines represents the "rate of change" of the Centiloid values. The blue lines, e.g. at 101, represent the expected trend of the Centiloid values increasing over time, while the red lines, e.g. at 102, represent the unexpected trend of the Centiloid values decreasing over time, which is a result of noise in the PET scan images. This unexpected trend of the Centiloid values is often overlooked, as the Centiloid value calculation usually occurs in the background during typical PET quantification. The noise that causes the unexpected trend in the Centiloid values may be due to noise in the PET images, noise from the reference region used to calculate the Centiloid values and changes to the PET tracer or scanner, for example.

[0053] Fig. 2 shows a plot of the rate of change of the Centiloid value verses the mean value of the Centiloid value. From the extended trend of the Centiloid values, there should not be any negative rates of change (i.e., each plot point should be above horizontal line 201). Fig. 2 also shows curve 202, which represents a calculated curve of mean value versus rate of change. Curve 202 may be obtained by calculating the mean value and rate of change for some or each piece of data in a set of longitudinal data. More specifically, curve 202 may be produced using a regression algorithm applied on the data of Fig. 2. It is expected that the data points of Fig. 2 should not deviate far from curve 202, given that curve is the expected trend of the data.

[0054] Given the expected trend of the longitudinal data (in this example, that the Centiloid values are expected to increase, such that the rate of change is positive), a training of a machine learning model can be designed to estimate a correction of the Centiloid values in such a way to penalise negative rates of changes, penalise over-correction and penalise distance from curve 202, for example. Training of the machine learning model is penalised in these ways by introducing constraints or penalty values which are formulated in a way to represent the penalisation of these factors.

[0055] While the disclosed system and method are advantageous for combating noisy ground truth data during training, there are also other advantages. For example, the volumes of structures present in training images may not be well defined (such as volume of white matter hyperintensity which have poor inter-rater reproducibility), which is compensated during training through the disclosed method. Another advantage is that the training can be performed using longitudinal data that is temporally sparse (taken far apart). This is particularly advantageous for training the machine learning model on medical images such as MRI or CT images, as data is only collected every 1 or 2 years. It is noted that the machine learning model can also be trained on less sparse data using the disclosed method, such as longitudinal data that collected every day or every week or every month, for example.

[0056] While this disclosure is generally directed towards the medical images and application of machine learning models in the medical field, it is noted that the disclosed system and method are not limited to applications in the medical field and medical images. More specifically, this disclosure relates more generally to training a machine learning model

on longitudinal data, where the longitudinal data may include medical images. However, in other instance, the longitudinal data may include other images. For example, any model for estimating a quantification/value from an image or any other input can be trained using the disclosed system and method, as long as there is some prior knowledge of the expected changes over time.

[0057] In other instances, the longitudinal data may not include images at all, but rather be some other form of data. For example, it may be expected that the rate of change of water flowing through a dam decreasing over time due to the predicted increase of droughts occurring in that area. However, it may be difficult to accurately measure the exact volume of water flowing through the dam, which introduces a source of noise. The volume may also fluctuate unexpectedly due to flash storms, which also introduces noise. The disclosed system and method can compensate for the noise in the data by utilising the expected rate of change of the data during training.

System for training

[0058] Fig. 3 illustrates an example system 300 for training a machine learning model on longitudinal medical scan images of a patient study. Fig. 3 is one example of a configuration of system 300. However, system 300 is not strictly limited to this configuration and this may be one possible embodiment of system 300. The medical scan images, or simply medical images, may be magnetic resonance (MR) image, positron emission tomography (PET) image or computed tomography (CT) image. Throughout this disclosure, medical scan images or medical images may be referred to simply as images or image data, noting that system 300 is equally applicable to images other than medical images.

[0059] A patient study is the study of a single patient where, for example, a medical image of the patient's brain is captured at regular time intervals, such as annually. It is noted that the machine learning model can be trained on longitudinal medical scan images of multiple patient studies, thereby being trained on data from multiple patients. This is because the penalty values indicative of an expected rate of change reduces the noise that results from inconsistencies in the data between patients. Machine learning model usually require large amounts of training data, so it is an advantage that the machine learning model can be trained

on longitudinal data from multiple patient studies. In embodiments outside of the medical field, the patient study may be a study of a subject over time.

[0060] System 300 comprises a device 301, which may be smartphone, computer or similar device. Device 301 comprises a processor 302 connected to a program memory 303 and a data memory 304. In other examples, device 301 is implemented in a distributed computing architecture, such as a cloud computing provider. Processor 302 may receive data through a variety of interfaces, which includes memory access of volatile memory, such as cache or RAM, or non-volatile memory, such as an optical disk drive, hard disk drive, storage server or cloud storage. The program memory 303 is a non-transitory computer readable medium, such as a hard drive, a solid state disk or CD-ROM.

[0061] Software, that is, an executable program stored on program memory 303 causes processor 302 to perform methods for training a machine learning model on longitudinal medical scan images of a patient study. For example, once executed, the software causes processor 302 to calculate two or more predictions of a biomedical factor by performing for each of two or more of the medical images: (i) applying the machine learning model to the medical image to calculate an output; and (ii) calculating a prediction using the output of the machine learning model for the medical image, calculate a rate of change between the two or more predictions for the two or more medical images and create a trained machine learning model by updating weights of the machine learning model to minimise a loss value.

[0062] The data memory 304 may store image data and retrieve the image data for later use. The image data may be indicative of a two-dimensional image, which may be stored on data memory 304 as Joint Photographic Experts Group (JPEG) format, RAW image format or a similar/equivalent image format, for example. In some examples, the image data may be an RGB image, a multispectral image, a hyperspectral image, an infrared image, or a two-dimensional cross-sectional. The image data may be indicative of a three-dimensional image, such as a PET, MR or CT image. Data memory 304 may store image data indicative of the three-dimensional image data as a single-file DICOM (Digital Imaging and Communications in Medicine), multi-file DICOM format or a similar/equivalent image format, for example.

[0063] The machine learning models described herein (such as the machine learning model before, during and after training) may be stored on data memory 304. The data memory 304 may also store predictions calculated by processor 302 performing any one of the disclosed methods, or any other variable or data necessary to perform such methods. For example, predictions, machine learning model outputs, loss values, rates of changes are calculated when processor 302 performs the disclosed method and can be stored on data memory 304.

[0064] It is to be understood that any receiving step may be preceded by processor 302 determining or computing the data that is later received. For example, processor 302 may then store the image data on data memory 304, such as on RAM or a processor register. Processor 302 then requests the data from the data memory 304, such as by providing a read signal together with a memory address. The data memory 304 provides the data as a voltage signal on a physical bit line and processor 302 receives the image data as the input image via a memory interface.

[0065] System 300 may also comprise a scanning machine 305, a server 306 and a monitor 307, which are each in communication with the device 301 via an input/output (I/O) port 308. In an example, the scanning machine 305 scans a patient by incrementally capturing two-dimensional cross-sectional images of the patient. The collection of cross-sectional images represents image data that is indicative of a scan and represents a 3D scan of the patient. Once scanning machine 305 creates the image data of the patient, processor 302 receives the image data via the I/O port 308. Processor 302 then performs the disclosed method using the image data received from the scanning machine 305 and trains the machine learning model.

[0066] If the image data is indicative of a MRI scan, then the image data may be created using an MRI scanning machine. MRI scanning machines consists of a table that slides into a cylinder. Inside the cylinder is a magnet that creates a strong magnetic field within the cylinder. The strong magnetic field is used to align the protons in the water molecules in a patient's body, as the protons possess a magnetic moment due to their intrinsic spin. Once the protons align with the strong magnetic field, a radio frequency signal is sent from the MRI scanning machine, which excites the protons and causes them to align against the strong magnetic field. Once the radio frequency signal is turned off, the protons relax and re-align

with the strong magnetic field, thereby emitting electromagnetic radiation, which is detected by the MRI scanning machine.

[0067] A patient lies on a motorised table that moves horizontally, in increments, through the cylinder and captures a single MR image at that increment, which corresponds to a two-dimensional cross-sectional image (virtual “slice”) of the patient’s body. Each two-dimensional cross-sectional image corresponds to a different region of the patient’s body as they move horizontally, in increments, through the cylinder. The collection of two-dimensional cross-sectional images also has an associated spatial sequence, in which the cross-sectional images are ordered along. The horizontal direction that the patient moves within the cylinder is indicative of a dimension in space along the spatial sequence of the two-dimensional cross-sectional images. The collection of two-dimensional cross-sectional images provides a three-dimensional representation of a patient’s internals, which aids physicians in diagnosing internal diseases and conditions.

[0068] The two-dimensional cross-sectional image consists of black and white regions, as well as shades of grey. These regions correspond to the rate at which excited protons return to their equilibrium state after being exposed to radio frequency (equilibrium state corresponds to the spin state which aligns with the strong external magnetic field), as well as the amount of energy released, changes depending on the environment and the chemical nature of the molecules. These factors can be differently weighted in the MR image, to give more contrast to certain tissues in the image. For example, in a T1-weighted (T1w) image, regions of high signal representing white regions may correspond to protein-rich fluid, whereas in T2-weighted (T2w) images, these regions may correspond to water rich fluids in the patient’s body.

[0069] If the image data is indicative of a CT scan, then the image data may be created using a CT scanning machine. CT scanning machines use a rotating X-ray tube and a row of detectors placed in a gantry of the machine to measure X-ray attenuations by different tissues inside the body. X-ray attenuations refer to when X-ray photons, with wavelengths ranging from 10 picometers to 10 nanometres, are absorbed by the tissue when the X-ray photons pass through the body of a patient.

[0070] A patient lies on a motorised table that moves horizontally, in increments, through the rotating X-ray tube and captures multiple X-ray measurements from different angles at each increment. The multiple X-ray measurements are then processed on a computer using reconstruction algorithms to produce tomographic two-dimensional cross-sectional images (virtual “slices”) of a body. Each two-dimensional cross-sectional image corresponds to a different region of the patient’s body as they move horizontally, in increments, through the rotating X-ray tube.

[0071] Similar to MR images, the collection of two-dimensional cross-sectional images also has an associated spatial sequence, in which the cross-sectional images are ordered. The horizontal direction that the patient moves within the cylinder is indicative of a dimension in space along the spatial sequence of the two-dimensional cross-sectional images. Similar to MR images, the collection of two-dimensional cross-sectional images provides a three-dimensional representation of a patient’s internals, which aids physicians in diagnosing internal diseases and conditions.

[0072] The two-dimensional cross-sectional (X-ray) images consists of black and white regions, as well as shades of grey. These regions correspond to the areas of the body that have attenuated the X-rays. For example, structures such as bones readily absorb X-rays and, thus, produce high contrast on the X-ray detector. As a result, bony structures appear whiter than other tissues in the X-ray image. Conversely, X-rays travel more easily through less radiologically dense tissues such as fat and muscle, as well as through air-filled cavities such as the lungs. These structures are displayed in shades of grey in the X-ray image. Regions where little to no attenuation has occurred will show up as black regions in the X-ray image.

[0073] If the image data is indicative of a PET scan, then the image data may be created using a PET scanning machine. Before the patient is placed into the PET scanning machine, a radiopharmaceutical (a radioisotope attached to a drug) is injected into the body as a radiotracer. When the radiopharmaceutical undergoes beta plus decay, a positron is emitted, and when the positron interacts with an ordinary electron present in the body, the two particles annihilate and gamma rays are emitted. These gamma rays are detected by gamma cameras to form a three-dimensional image, in a similar way that an X-ray image is captured.

[0074] Some parts of the body absorb more of a particular radiotracer than other parts of the body. For example, diseased cells, such as cancer cells, in a patient's body absorb more of the radiotracer than healthy ones do. In another example, the brain is normally a rapid user of glucose, where radiotracer may be glucose containing the radioisotope. Areas where there is a higher detection of gamma ray (as referred to as "hot spots") will appear on the PET images as hotter colours, such as yellow and red, as opposed to areas of lower gamma ray detection, which will appear as blue. As such, the size and location of these "hot spots" enable the diagnosis of disease from the PET images.

[0075] Processor 302 may also receive other longitudinal data from other sources. For example, system 300 may comprise a RGB camera (not shown) configured to capture RGB image data intermediately over a period of time. In another example, system 300 may comprise an inertial measurement unit (IMU) which measure position and pose data of a machine intermediately over a period of time. Moreover, system 300 may comprise a set of sensors which provide measurements to processor 302 intermediately over a period of time. As such, the disclosed system and method are not limited to applications involving medical images.

[0076] The image received from the scanning machine 305 may be stored on the data memory 304 before and after the disclosed method is performed by processor 302. In another example, the server 306 may store image data. Additionally, image data may be communicated to the server 306 and the server 306 may then communicate the image data to device 301. The image data may be received from a source external to system 300, such as another system located remotely to system 300. The remote system may be located within the same facility as system 300 or may be completely remote from system 300. The image data may be stored in the server 306 as a single-file DICOM or multi-file DICOM format.

[0077] The server 306 may be in communication with the scanning machine 305, or may be in communication, over a communications network. Similarly, processor 302 receives the image data via the input/output port 308 and performs the disclosed method on the image data. After processor 302 performs the disclosed method on the image data, the trained machine learning model may be communicated to the server 306 by communicating the

parameters of the trained machine learning model. The server 306 may then communicate these parameters to an external system or store them.

[0078] System 300 may also comprise a monitor 307, which may be configured to display the image data from the scanning machine 305 and from the server 306. In this way, processor 302 communicates the image data from the scanning machine 305 and from the server 306 to the monitor via the input/output port 308. Monitor 307 may be further configured to the result of applying the trained machine learning model of a new medical image during inference. In this way, processor 302 communicates the inference result to the monitor via the I/O port 308.

[0079] While only one device 301 is depicted in Fig. 3, there may be a network of many devices that may communicate with the server 306. The network of devices may also communicate directly with one another via the Internet, any other means of wireless communication or wired connections. In an example, multiple devices may each capture image data of a patient. Each instance of image data from the multiple devices may then be stored on server 306 for later use. In some embodiments, a database of image data captured from multiple devices may be stored on server 306.

[0080] It is noted that server 306 may have similar functionality to that of device 301. Server 306 may perform parts of the disclosed methods described herein. For example, program memory may comprise software, that is, an executable program stored on program memory, and causes the processor to train a machine learning model on longitudinal data.

[0081] Software may provide a user interface presented to the user on device 301. The user interface is configured to accept input (via buttons or text fields etc.) from the user, via a touch screen or a device attached to device 301 such as a keyboard or computer mouse. These devices may also include a touchpad, an externally connected touchscreen, a joystick, a button, and a dial. In an example, device 301 may display multiple instances of image data, and the user may choose one of the multiple instances of image data such as medical images, by processor 302. The user may choose one of the multiple instances of image data by interacting with the touch screen or inputting the selection with a keyboard or computer mouse.

[0082] Processor 302 may receive or send data, such as image data, from data memory 304 as well as from the I/O port 308. In one example, processor 302 sends image data from the device 301 to server 306 via I/O port 308, such as by using a Wi-Fi network according to IEEE 802.11. The Wi-Fi network may be a decentralised ad-hoc network, such that no dedicated management infrastructure, such as a router, is required or a centralised network with a router or access point managing the network. System 300 may further be implemented within a cloud computing environment, such as a managed group of interconnected servers hosting a dynamic number of virtual machines.

[0083] Although I/O port 308 is shown as single entity, it is to be understood that any kind of data port may be used to receive data, such as a network connection, a memory interface, a pin of the chip package of processor 302, or logical ports, such as IP sockets or parameters of functions stored on program memory 303 and executed by processor 302. The parameters of functions may be stored on data memory 304 and may be handled by-value or by-reference, that is, as a pointer, in the source code.

Method for training

[0084] Fig. 4 illustrates method 400 for training a machine learning model on longitudinal medical scan images of a patient study. Fig. 4 is to be understood as a blueprint for the software program and may be implemented step-by-step, such that each step in Fig. 4 is represented by a function in a programming language, such as Python, C++ or Java. The resulting source code is then compiled and stored as computer-executable instructions on program memory 303, which causes processor 302 to perform method 400. Before the process of training the machine learning model begins, processor 302 may initialise the weights of the machine learning model (which define the machine learning model). For example, processor 302 may use random weights at the beginning of the train process, which are then updated during the training process.

[0085] In some examples, the machine learning model may be a neural network. In further examples, the machine learning model may be a neural network comprising one or more convolutional layers. Such a machine learning is known as a convolutional neural network (CNN). The CNN is ideal for applications involving images as it accounts for the positioning

and shape of objects in the image data. However, other types of machine learning models are equally applicable here. For example, the machine learning model may be implemented as a K nearest neighbour model, decision tree and support vector machine.

[0086] Mathematically, a convolution is an integration function that expresses the amount of overlap of one function g as it is shifted over another function f . Intuitively, a convolution acts as a blender that mixes one function with another to give reduced data space while preserving the information. A CNN performs convolutions with filters (matrix / vectors) and the filters comprise learnable parameters that are used to extract low-dimensional features from an input data. They have the property to preserve the spatial or positional relationships between input data points. CNNs exploit the spatially-local correlation by enforcing a local connectivity pattern between neurons of adjacent layers. In this example, the CNN architecture may comprise an optimal number of convolutional layers, filter size and stride length.

[0087] Intuitively, a convolution is the step of applying the concept of a sliding window (a filter with learnable weights) over the input and producing a weighted sum (of weights and input) as the output. The weighted sum is the feature space which is used as the input for the next layers. More specifically, each convolutional layer comprises a filter, which calculates a weighted sum of adjacent pixel values. For example, a 2 x 2 filter comprises 4 weights, which are the coefficients of the filter. The filter starts at an initial position in the image data structure, multiplies each pixel value in the image data structure with the respective filter coefficient and adds the results. Finally, the filter stores the resulting number in an output pixel. In this sense, the output pixel value of each of the one or more convolutional layers comprises a weighted sum of input pixel values. The weights in the weighted sum correspond to the coefficients of the filter. Then, the filter moves by one pixel along one direction in the data structure and repeats the calculation for the next voxel of the output image. For example, if the stride is 2, the filter will move by two pixels along one direction. That direction may be an x-dimension or a y-dimension.

[0088] A single convolutional layer may use multiple filters, where each filter corresponds to a different feature. The corresponding feature of each filter is determined during training of the CNN. More specifically, each filter may contain weights, which are adjusted during a training process through a backpropagation process, for example. The filters are used to

quantitatively determine the contribution of a particular feature on the output of the CNN. For a convolutional layer that occurs at the start of a CNN architecture, the filters of this convolutional filter may correspond to simple features. However, convolutional layers that occur later in the CNN architecture may exhibit more complex or abstract features. As an example, the complex or abstract features may be a combination of the simple features from the previous convolutional layer.

[0089] The CNN architecture may also comprise a batch normalization layer and rectifier linear unit (ReLU) activation function. Further, the output of the last convolution layer may be subjected to Global Average Pooling (GAP). Even further, a fully connected layer with Sigmoid activation may also be used on the output layer. However, other activation functions may be used. These activation functions include, but are not limited to, a binary step function, a tanh function, a ReLU function or a softmax function.

[0090] Fig. 5 is an example CNN architecture 500. This example CNN architecture 500 is designed to predict parameters from an MR image for dynamically transformation of the intensity contrast of an input MR image to adapt to a segmentation task. More specifically, the example CNN predicts the parameters for a power function transformation and a piece-wise linear transformation. As can be seen in Fig. 5, the example CNN architecture 500 includes three convolutional blocks 511, 512, 513 and three fully connected (FC) blocks 521, 522, (523 and 524 are collectively one FC block). This CNN architecture 500 is used to provide results for the disclosed method presented later in the disclosure.

[0091] To begin method 400, processor 302 may select 401 two or more images from the longitudinal data of a patient study. Preferably, the two or more images come from the same patient, rather than the two or more images being from different patients. The two or more images may be randomly chosen from the patient study. For example, the two or more images do not have to be images that have been captured consecutively.

[0092] Processor 302 then calculates 420 two or more predictions of a biomedical factor each of two or more of the medical images. The biomedical factor has an expected trend that is known beforehand and is utilised to train the machine learning model. For each of the two or more of the medical images, processor 302 applies 421, 426 the machine learning model to

the medical image to calculate 422, 427 an output, then calculates 423, 428 a prediction using the output of the machine learning model for the medical image. Preferably, processor 302 calculates 420 the two or more predictions using the same machine learning model and more specifically, processor 302 calculates 420 the two or more predictions using a model that has the same architecture and weights.

[0093] In some embodiments, processor 302 calculates 420 two or more predictions of a factor that is not a biomedical factor. The factor may be any variable that has some expected trend (e.g., expected to change over time). For example, the factor may be a physical factor such as the volume of water in a dam. It is noted that while this disclosure describes embodiments related to a biomedical factor, these embodiments are equally applicable to a factor other than a biomedical factor.

[0094] It is also noted that the calculation of the prediction for each of the two or more medical images does not have to be performed sequentially, in the sense that the first prediction is calculated then the second prediction is calculated and so on. The prediction for each of the two or more medical images may be calculated in parallel, as the calculations are not necessarily dependent on each other. For example, each prediction may be calculated in parallel across many CPUs and multiple cores. More specifically, each prediction may be calculated on a single CPU efficiently by multi-platform shared-memory parallel programming, such as OpenMP, which utilises the multiple computing cores of the CPU.

[0095] In some embodiments, processor 302 calculates 423, 428 the prediction by applying a function to the output of the machine learning model. For example, the function may be a polynomial, exponential or trigonometric function or combination thereof. In other embodiments, processor 302 calculates 423, 428 the prediction by multiplying the output by a factor or measurement. For example, the output may be a scaling factor to correct a specific measurement and hence, the prediction corresponds to the corrected measurement. In further embodiments, the output may simply correspond to the prediction, which can be considered as processor 302 calculating 423, 428 the prediction by multiplying the output by 1.

[0096] The biomedical factor may be one of medical image quantification factor, a biological structure value or a biomarker. For example, the biomedical factor may be a

medical image quantification factor, which may be a Centiloid value or standardised uptake value ratio scaling factor. In other examples, the biomedical factor may be a biological structure value, which may be a volume of a biological structure visible in the medical image, such as the grey matter in the brain. The biomarker may be blood pressure or heart rate, for example.

[0097] After calculating 420 the two or more predictions, processor 302 calculates 430 a rate of change between the two or more predictions for the two or more medical images. In some embodiments, processor 302 calculates 430 the rate of change based on a difference between the two or more predictions. In other embodiments, processor 302 calculates 430 the rate of change based on a difference between the two or more predictions and a difference between a variable on which the rate of change is based. For example, if there are only two predictions, processor 302 may calculate the rate of change using a difference of the predictions divided by the difference between the timepoints of each respective medical image.

[0098] The calculated rate of change is indicative of a trend in the longitudinal data. As such, processor 302 may calculate the rate of change explicitly by calculating a ratio of the change in the predictions and the change in an independent variable (such as time). However, processor 302 may also calculate the rate of change implicitly by calculating a difference between the two or more predictions, as the medical images may inherently contain change information (such as temporal information). For example, two medical images in a patient study may be taken a year apart and hence, the two medical images have inherent temporal information, given that they are temporally related. As such, the rate of change may be calculated by a difference of the two predictions of the respective medical images, as the medical images inherently contain temporal information.

[0099] After calculating 430 the rate of change, processor 302 calculates 440 a loss value to train the machine learning model. In particular, the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change. Reflecting a difference generally means that the penalty value is indicative of the difference. Reflecting a difference may mean that the penalty value is related to the difference, such as being directly proportional to the difference. Reflecting a difference may also mean that processor 302

calculates the penalty value using the difference, such as by applying a mathematical function to the difference. By calculating the loss value using this penalty value, training of the machine learning model leverages information that is not available during inference.

[0100] In some examples, the expected rate of change may be a function of time (such as linear, polynomial, exponential, etc.) or may be a second order rate (i.e., the rate of the rate of change) and processor 302 calculates the second order rate of change. In other examples, the expected rate of change may be zero, indicating that the biomedical factor remains constant over time. As such, the penalty value may be formulated in such a way to penalise the rate of change deviating from zero.

[0101] Calculating a loss value may be referred to as applying a loss function. In other words, the outcome of applying a loss function is the loss value. The loss value and loss function may also be referred to as cost value and cost function, respectively. The loss function may be applied to the outputs and other factors, such as the timestamps of two or more medical images. The loss value may incorporate many types of additional information that is not available at inference time. For example, the difference curves based on clinical diagnostic, age or any other relevant information may be used to train the machine learning model using the loss value.

[0102] Processor 302 then updates 450 weights of the machine learning model to minimise a loss value, thereby creating a trained machine learning model. This process is generally referred to as “training”. The longitudinal data (i.e., the training data) comprises label data, such as image data that has been labelled with a class, where the label for each item of training data may be manually provided by a user. Training data may also or otherwise be obtained from a database containing already labelled data, for which manually labelling the data is not needed. Training the machine learning model in this way is referred to as “supervised learning”, in the sense that the outcome of the machine learning model is validated and the weights of the machine learning model are adjusted accordingly. The weights may be updated through the process of backpropagation. The weights may also be updated through a gradient descent process.

[0103] It should be noted that “supervised learning” is not the only method that can be used to train the machine learning models. For example, “semi-supervised learning” can be used in situations where only a relatively small amount of training data can be obtained. In some examples, a training set may be augmented to produce more training data, which is advantageous if a large training data is difficult to obtain. A training set may be augmented by randomly resizing, cropping, flipping, or rotating the image data in the training set, for example.

[0104] It is noted that during method 400, the rates of change (e.g., the rate of change between the two or more predictions) may only be used during training to update the machine learning model, rather than at inference time as the prediction provided by the machine learning model may correspond to single timepoint measurements. As such, the rates of change may not be used at inference time to obtain the prediction from the trained machine learning model. Rather, single timepoint measurements may be used at inference time to generate corresponding predictions using the trained machine learning model.

Repeating the training method

[0105] Method 400 is preferably repeated multiple times using two or more different images. In general, the more repetitions of method 400, the more accurate the machine learning model becomes at making a prediction during inference. Moreover, training the machine learning model on a range of different images or training data also generally makes the machine learning model more accurate at making a prediction during inference. In this disclosure, reference to “trained machine learning model” means that the initial machine learning model has undergone at least one iteration of method 400. However, “trained machine learning model” may also mean that the initial machine learning model has undergone multiple iterations of method 400. Other terms such as “further trained machine learning model”, “another trained machine learning model” and “fully trained machine learning model” may also be used throughout this disclosure to indicate that the machine learning model has undergone multiple training iterations.

[0106] As processor 302 may perform method 400 multiple times using two or more different images, processor 302 may calculate a further two or more predictions of the

biomedical factor by, for each of a further two or more medical images of the patient study: (i) applying the trained machine learning model to the medical image to calculate an output; and (ii) calculating a prediction using the output. Processor 302 may then calculate a rate of change between the further two or more predictions; and creates a further trained machine learning model by updating weights of the trained machine learning model to minimise a further loss value based on the further rate of change. As such, the machine learning model is trained on different images of the same patient study. This process may be repeated using each permutation of the two or more images in the patient study.

[0107] In some embodiments, processor 302 may further train the machine learning model using images from another patient study, which is different to the patient study used to initially train the machine learning model. More specifically, processor 302 may calculate another two or more predictions of the biomedical factor by, for each of two or more medical images of another patient study different to the patient study: (i) applying the trained machine learning model or further trained machine learning model to the medical image to calculate an output; and (ii) calculating a prediction using the output. Processor 302 may then calculate another rate of change between the other two or more predictions; and creates another trained machine learning model by updating weights of the trained machine learning model or further trained machine learning model to minimise another loss value based on the other rate of change.

[0108] The machine learning model may be agnostic to the order of the two or more medical images of the patient study. As such, the machine learning model is unable to learn the order as each image is processed the same way, with the only difference being how the loss is computed. It is beneficial for the machine learning model not to learn the order of the input images, as it is desired that the model learns the expected trend in the data (i.e., the expected rate of change), rather than the sequence of the images. This provides better predictions during inference.

[0109] To further re-enforce this, the order of the inputs (i.e., the two or more medical images) can be randomly switched. As such, in some embodiments, each of the medical images of the patient study has a timestamp and at least two of the two or more medical image have non-adjacent timestamps. Non-adjacent timestamps being non-consecutive. For

example, if the patient study contained longitudinal data of a brain CT image captured every year, then the non-adjacent timestamps would be the non-consecutive years.

Calculating the expected rate of change

[0110] As the penalty reflects a difference between the rate of change and the expected rate of change, processor 302 may calculate the expected rate of change. Processor 302 may calculate the expected rate of change from the longitudinal data. For example, processor 302 may calculate a rate of change for each pair of images in the patient study. Processor 302 may then average the calculated rates of change to calculate the expected rate of change. Processor 302 may also use other techniques such as regression to calculate the expected rate of change from the longitudinal data.

[0111] In some embodiments, processor 302 determines a polynomial function of rate of change of the biomedical factor and calculates the expected rate of change using the polynomial function. This polynomial function represents the expected trend of the longitudinal data and hence, can be used to calculate the expected rate of change of the biomedical factor based on the two or more medical images used to train the machine learning model. For example and with reference to Fig. 2, curve 202 represents a polynomial function and is based on the longitudinal data used to train the machine learning model. It is noted that this function is not limited to a polynomial function. While preferably a polynomial function, this function may be a polynomial, exponential or trigonometric function or combination thereof.

[0112] In some embodiments, processor 302 calculates the expected rate of change using the polynomial function by applying the polynomial function to a mean value of the biomedical factor calculated by an average of the two or more predictions. Specifically, the mean value may be calculated using a difference of the two or more predictions divided by the number of two or more predictions. The polynomial function may be a rate of change verses mean value function and hence, may be a function of the mean value that, when applied to the mean value, results in the rate change corresponding to that mean value.

[0113] In some embodiments, processor 302 determines the polynomial function by calculating a rate of change and a mean value for each pair of medical images in the patient study. Moreover, processor 302 may determine the polynomial function by calculating a rate of change and a mean value for each pair of medical images in each patient study that form the training data. Processor 302 may determine the polynomial function by determining the best representation of the training data. For example, processor 302 may use regression (e.g., curve of best fit) or other techniques, such as applying a machine learning model.

Further penalty values

[0114] In some embodiments, the loss value further comprises a change penalty value reflecting a difference between the two or more predictions, where the change penalty value penalises an unexpected temporal trend of the biomedical factor related to the expected rate of change. For example, the expected temporal trend of the longitudinal data may be a positive rate of change over time. As such, the change penalty value would penalise instances of a negative rate of change occurring. In some examples, the change penalty value may comprise a clamp function to only keep positive or negative changes.

[0115] In some embodiments, the loss value further comprises a correction penalty value reflecting a difference between the two or more predictions and two or more corresponding ground truth values. The correction penalty value may penalise the two or more predictions from deviating from the two or more corresponding ground truth values. For example, the correction penalty value may comprise two penalty values reflecting the slope and intercept between the two or more predicted values and the ground truth measurements. As such, the correction penalty value may be based on a difference between the rate of change of the two or more predictions and the rate of changes of the two or more corresponding ground truth values. In other examples, the slope and intercept may be based on a regression line between the ground truth measurements and the two or more predictions.

[0116] In some examples, it may be expected that the slope between output of the predicted values and the ground truth measurements should not deviate too far from 1. In some examples, it may be expected that the intercept between output of the predicted values and the ground truth measurements should not deviate too far from 0. As such, over-correction may

be penalised during the training of the machine learning model using the two correction penalty values that reflect this expectation. This may ensure that no bias is introduced during the training process. In some examples, the correction penalty value may be a sum of the differences between each output and the over-correction factor. In further examples, the correction penalty value may be a sum of the absolute value of these differences.

[0117] For example, a regression line may be calculated by determining a slope and intercept based on each prediction and the corresponding ground truth value. More specifically, a plot may be made with data points corresponding to the prediction and the corresponding ground truth value, then a linear regression of the plot may be determined. In essence, a linear function may be calculated where the input to the linear function corresponds to a prediction and the output (i.e., value of the function evaluated at an input) corresponds to a ground truth value. Hence, the slope may be given as:

$$\text{slope} = \frac{\text{ground truth}}{\text{prediction}}$$

[0118] As it is desired that the prediction similar to the ground truth, it may be logical that the slope of this linear function should be around 1, while the intercept of the linear function should be around 0. Hence, the correction penalty may be based on a difference between the slope and 1, as well as the intercept and 0, such that this difference is minimised during training. For example, the correction penalty value may be given by:

$$\text{correction penalty value} = \text{abs}(\text{slope} - 1) + \text{abs}(\text{intercept} - 0)$$

[0119] However, it is noted that other variations of the correction penalty value may be possible.

[0120] In some embodiments, the loss value further comprises a correction penalty value reflecting a difference between the two or more outputs and an over-correction factor, where the correction penalty value penalises the two or more outputs from deviating from the over-correction factor. In some examples, it may be known that the output of the machine learning model should not deviate too far from an over-correction factor. As such, over-correction can

be penalised during the training of the machine learning model using the correction penalty value. In some examples, the correction penalty value may be a sum of the differences between each output and the over-correction factor. In further examples, the correction penalty value may be a sum of the absolute value of these differences.

[0121] In some embodiments, the loss value further comprises an additional penalty value calculated by one or more of: (i) a difference between each of the two or more outputs of the machine learning model and a respective ground truth measurement, and (ii) a mean square error between each of the two or more outputs and the respective ground truth measurement. As such, the loss value may comprise additional (standard) loss terms. In essence, the loss value may comprise any number of additional loss terms.

[0122] In some embodiments, processor 302 calculates the loss value by a weighted sum. For example, the loss value may comprise the penalty value, the change penalty value, the correction penalty value, and the additional penalty value. As such, processor 302 calculates the loss value by a weighted sum of the penalty value, the change penalty value, the correction penalty value, and the additional penalty value. The weights of the weighted sum may be considered “learning rates”, which are predetermined before training occurs, such as by user input. In some examples, one or more of the weights may be 1.

Aiding in diagnosis

[0123] It is noted that the biomedical factor can then be used to aid in determining a diagnosis for the patient. For example, a clinician may apply the trained machine learning model to an image of a new patient. As such, method 400 may further comprised applying the trained machine learning model to a medical image of a patient to aid in diagnosis of the patient based on the output of the trained machine learning model.

[0124] In one example, the trained machine learning model may calculate a measure, e.g., amount of amyloid plaques on the brain. The method may then apply one or more thresholds on that calculated measure to determine that the patient has Alzheimer’s pathology or that there is a low/medium/high risk of developing Alzheimer in the future. This use of the trained

machine learning model may equally be possible for other medical indications, including progressive diseases, or general phenotypes.

[0125] Diagnosis and the study of Alzheimer's disease is often performed using PET quantification of a patient's brain PET image. The PET quantification may provide quantification information such as amount of amyloid plaques in the brain, for example. PET quantification may also involve processor 302 creating a three-dimensional rendering of the patient's brain from the PET image. Hence, in some embodiments, each of the medical images is a PET image and method 400 comprises performing PET quantification by applying the trained machine learning model to a PET image of a patient.

[0126] As such, to aid in the diagnosis of Alzheimer's disease, the biomedical factor would be a medical image quantification factor and more specifically, the medical image quantification factor would be a Centiloid or standardised uptake value ratio scaling factor. Moreover, to aid in the diagnosis of Alzheimer's disease, method 400 would further comprises applying the trained machine learning model to a medical image of a patient to aid in diagnosis of Alzheimer's disease based on the Centiloid or standardised uptake value ratio scaling factor that is an output of the trained machine learning model.

Example embodiment

[0127] An example embodiment of the disclosed method will now be presented for explanatory purposes but is not intended to limit the disclosed system and method to this embodiment. This example embodiment is also illustrated with experiment results of training a machine learning model using the disclosed method.

[0128] In this example embodiment, the biomedical factor is a medical image quantification factor and more specifically, the biomedical factor is a Centiloid value. The output of the machine learning model being trained is a scaling factor for the SUVR value, before it is transformed into Centiloid value, and the predictions are scaled Centiloid values. As such, the medical images used during training and inference will be PET images, as the SUVR and Centiloid scale relate to PET images.

[0129] Centiloid values are used to standardise amyloid PET images for PET quantification by normalising the PET SUVR across different amyloid PET tracers to a standard and common unit system, with each PET tracer having its own linear transform to normalise SUVR into Centiloid. However, due to noise in the PET images and the use of different PET tracers/scanners which can have different levels of noise and bias, the Centiloid values are often noisy. Therefore, PET quantification that relies on PET images in the Centiloid scale may be less accurate as a result of the noise in the PET image and the variability introduced by the use of different tracers and scanners. As such, machine learning techniques can be used to determine a correction to the calculated SUVR value, before they are transformed into Centiloid value, in an effort to reduce the effect of noise in the PET images and increase the accuracy of the PET quantification.

[0130] In this example embodiment, the machine learning model is trained on two medical images at a time (i.e., two images from the same patient study for each training iteration). As such, processor 302 calculates two predictions of the Centiloid value (i.e., scaled Centiloid value) by applying the machine learning model to each of the two medical images, individually, calculating a respective output and calculating a respective prediction using the respective output.

[0131] In this embodiment, processor 302 calculates the predictions by first calculating the SUVR value for each of the two medical images. Processor 302 may calculate the SUVR value by calculating a ratio between a reference region of the respective PET image and a “hot spot” region of the respective PET scan, such as an amyloid, before transforming them into Centiloid. Processor 302 may also use the tracer and/or scanner information as part of the prediction, so that there are different corrections for different tracers, which allows to better control for their different levels of noise. Processor 302 then calculates the predictions by a product of the calculated SUVR value and the output of the machine learning model (i.e., the scaling factor), before transforming the scaled SUVR into Centiloid (i.e., the scaled Centiloid value).

[0132] Processor 302 then calculates a rate of change between the two predictions by calculating a difference between the two predictions (i.e., the scaled Centiloid value) divided by a difference between the timestamps of the two medical images, given by:

$$\text{rate of change} = \frac{\text{prediction}_2 - \text{prediction}_1}{\text{timestamp}_2 - \text{timestamp}_1}$$

[0133] Processor 302 then calculates the loss value. It is noted that in some embodiments, processor 302 calculates the loss value by applying a loss function. For example, in the example embodiment, processor 302 applies a loss function to the calculated Centiloid values, the outputs of the machine learning model and the timestamps of the two medical images. In other words, the calculated Centiloid values, the outputs of the machine learning model and the timestamps of the two medical images are inputs into the loss function. As such, processor 302 may calculate the rate of change by applying the loss function to the inputs.

[0134] In this example embodiment, the loss value comprises three penalty values: (i) the penalty value indicative of the rate of change, (ii) the change penalty value penalising an unexpected temporal trend of the Centiloid value, and (iii) the correction penalty value penalising the two or more outputs from deviating from the over-correction factor. In this embodiment, processor 302 calculates the loss value using a weighted sum of the three penalty values. For the experimental results, the loss value was calculated using:

$$\text{loss value} = \text{change penalty value} + \alpha * \text{correction penalty value} + \beta * \text{penalty value}$$

where α and β are pre-determined parameters or learning rates.

Penalty value indicative of the rate of change

[0135] To calculate this penalty value, processor 302 calculates a rate of change and a mean value for each pair of medical images for each patient study. Processor 302 then determines a polynomial function by applying a regression algorithm to the calculated rates of change. The calculated rates of change and the determined polynomial function for this example embodiment are shown in Fig. 2. In essence, it is expected that the rate of change does not deviate from the polynomial. As such, this penalty value indicative of the rate of change penalises large distances away from the polynomial.

[0136] Processor 302 calculates the expected rate of change using the polynomial function by applying the polynomial function to a mean value of the Centiloid value calculated by an average of the two or more predictions. More specifically, the mean value is given by:

$$\text{mean value} = \frac{\text{prediction}_2 - \text{prediction}_1}{2}$$

[0137] Processor 302 then calculates this penalty value by calculating a difference between the calculated rate of change and the expected rate of change. More specifically, processor 302 calculates the absolute value of the difference between the calculated rate of change and the expected rate of change. As such, the penalty value is given by:

$$\text{penalty value} = \text{abs}(\text{calculated change} - f(\text{mean value}))$$

where $f(\cdot)$ is the polynomial function.

Change penalty value

[0138] The change penalty value penalises an unexpected temporal trend of the Centiloid value related to the expected rate of change. In the example embodiment, it is expected that Centiloid value will increase over time and hence, the rate of change of the Centiloid value is expected to be positive (more than zero). As such, the change penalty value is formulated in such a way to penalise decreasing Centiloid values and negative rates of change. In the example embodiment, the change penalty value is based on a “baseline/follow up” concept, where patients will have a “baseline” medical image scan performed which will be compared to a “follow up” medical image scan in the future.

[0139] The change penalty value is given by:

$$\text{change penalty value} = \text{clamp}(\text{prediction}_2 - \text{prediction}_1, 0)$$

where “clamp” is the clamp function, which restricts a given value between an upper and lower bound. More specifically, the clamp function is defined by:

$$\text{clamp}(x) = \max(a, \min(x, b)) \in [a, b]$$

[0140] The clamp function restricts the difference in the predictions to only be positive, thereby enforcing the extended trend of an always positive rate of change during the training.

Correction penalty value

[0141] The correction penalty value penalises the two or more outputs of the machine learning model from deviating from an over-correction factor. In the example embodiment, the over-correction factor is 1, as the output of the machine learning model is a scaling factor and hence, the output should be ideally 1 in the absence of noise in the PET image.

[0142] The correction penalty value is given by:

$$\text{correction penalty value} = \text{abs}(\text{output}_1 - 1) + \text{abs}(\text{output}_2 - 1)$$

Results

[0143] The results of training the machine learning model are shown in Fig. 6. To produce these results, the machine learning model was trained using 5,000 PET images from 1,500 patients. The machine learning model used to generate these results was a CNN of the architecture shown in Fig. 5, noting that the input to the machine learning model is a PET image rather than an MR image. Fig. 6 shows the progression of the calculated rate of change of the scaled Centiloid values as the training of the machine learning model increases. In particular, it can be shown that the calculated rates of change move towards the polynomial function as the training continues. Further, the calculated rates of change become more positive as per the expected rate of change of the Centiloid values.

Generating masks for machine learning explainability

[0144] Machine learning models (particularly deep machine learning models) are often known for being black boxes, meaning that it is unknown how they calculate a prediction from an input. For example, while the models may be able to detect the presence of a disease

from image data, it may not be clear how the model makes its prediction nor may it be clear what features the model deems important when making its prediction. The black box nature of machine learning model means that interpretability can be challenging. The understanding of decisions made by black-box algorithms is a challenge and assessing their fairness and unbiasedness is an important step to deployment in many industries.

[0145] As previously discussed, the trained machine learning model disclosed herein may output a prediction of a biomedical factor, which may be a standardised uptake value ratio (SUVR) scaling factor (which may be simply referred to as a SUVR). Recall that the SUVR may be defined as the ratio of a region of the PET image containing specific binding (a target region where the PET tracer will bind to the target), and a region of the PET image containing non-specific binding (as referred to as a reference region). As such, a target mask may be generated which highlights the target regions in a PET image. Similarly, a reference mask may be generated which highlights the non-specific binding regions in a PET image. In the case of Centiloids in a PET image, the reference mask may be referred to as a Centiloid reference mask or the like. Hence, generating masks that correspond to the corrections provided by the trained machine learning model may provide insight into which regions of the PET image that the trained machine learning uses in its determination of corrected SUVR.

[0146] It is thought that the corrections provided by the trained machine learning model disclosed herein may reflect changes in either the reference or target mask, or both. This idea may be leveraged to generate masks with SUVRs corresponding to the corrections provided by the trained machine learning model. In other words, new masks may be optimised, for each tracer, such that the correlation with the SUVR values provided by the trained machine learning model is optimised (e.g., maximised). This may provide insight into the features deemed important to the trained machine learning model, which may not be obvious in the original PET image, for example.

[0147] As such, in some embodiments, processor 302 may generate one or more masks that correspond to corrections determined by the trained machine learning model. It is noted that a “mask” in the present context may refer to an image, such as a heat map, saliency map or the like, that highlights relevant regions for machine learning model decision making. Processor 302 may generate the one or more masks by adapting one or more medical images (that may

be input into the trained machine learning model) based on a corresponding output of the trained machine learning model. For example, processor 302 may apply an algorithm, mathematical operation or a further machine learning model to the one or more medical images based on the corresponding output to generate the one or more masks.

[0148] Obtaining these new masks may be formulated as an optimisation problem where, for each tracer, a new reference and target mask may be optimised across the entire dataset so that the resulting SUVR maximises the Pearson correlation coefficient with the corrected SUVRs provided by the trained machine learning model. The Pearson correlation coefficient is a correlation coefficient that measures linear correlation between two sets of data.

[0149] This optimisation problem may be solved using a gradient descent approach or another type of optimisation technique. The lost function may involve the following two losses:

1. **Penalises low Pearson correlation coefficient.** Using the updated masks, new SUVRs can be computed across the entire dataset. The correlation between these SUVRs and the Corrected SUVR obtained by the trained machine learning model are used to compute the Pearson coefficient R^2 . The resulting loss is defined as:

$$L_p = 1 - R^2$$

2. **Promotes binary masks.** The two masks being optimised need to have continuous values to ensure that they can be derived throughout the optimisation process. To force the masks towards binarity, for each mask M , values not being either 1 or 0 are penalised as follow:

$$L_b = \sum 0.5 - |M - 0.5|$$

[0150] Experiment and results

[0151] An experiment was conducted to generate reference and target masks for machine learning model explainability. The experimental details are now provided. However, it is noted that alternatives and alterations are possible in the generation of these masks.

[0152] The reference and target masks were initialised using the original Centiloid reference and target mask. The masks were then smoothen using a Gaussian kernel, specifically a 4mm FWHM Gaussian kernel. At each iteration, the masks were updated using the gradient information, and smoothed again using the same Gaussian kernel to reduce sparseness. The masks were also be mirrored to generate symmetric masks. In this experiment, the optimisation leveraged pytorch's autograd engine for automatic computation of the gradients.

[0153] The optimisation was stopped once the Pearson loss stopped to improve anymore for 10 epochs or if the maximum number of epochs (2000) was reached. Once the final masks were obtained, the new SUVRs derived from these masks were transformed into the Corrected SUVR using a linear transform, before being transformed into Centiloids using the standard transforms.

[0154] Fig. 7 shows the result of this optimisation process. In particular, Fig. 7 shows the progression of the SUVR derived from the optimised masks as the optimisation processes continues (i.e., after a number of iterations). It can be seen in Fig. 7 that, as the optimisation process continues, the ratio of the Corrected SUVR provided by the trained machine learning model (labelled as DeepSUVR SUVR on the x-axis) and the SUVR determined from the optimised masks approaches 1 (as indicated by the datapoints approaching the diagonal line). This indicates that through the optimisation process described above, the reference and target masks are indeed optimised such that the SUVR determined from these masks correlates to the correspond SUVR provided by the trained machine learning model.

[0155] Fig. 8a shows the initialised target masks, while Fig. 8b shows the optimised target masks. Similarly, Fig. 9a shows the initialised reference masks, while Fig. 9b shows the optimised reference masks. It is noted that for both the reference and target masks, the optimised masks still include salient features that can be seen in the initialised masks. However, the optimisation process provides more salient features and better definition of these features. These results provide a better indication of the features considered important

by the trained machine learning model disclosed herein in making its prediction. As such, these results aid in the explainability of the trained machine learning model. These optimised masks can also potentially replace the machine learning model for directly generating Corrected SUVRs.

[0156] It will be appreciated by persons skilled in the art that numerous variations and/or modifications may be made to the above-described embodiments, without departing from the broad general scope of the present disclosure. The present embodiments are, therefore, to be considered in all respects as illustrative and not restrictive.

CLAIMS:

1. A method for training a machine learning model on longitudinal medical scan images of a patient study, the method comprises:
 - calculating two or more predictions of a biomedical factor by performing for each of two or more of the medical images:
 - applying the machine learning model to the medical image to calculate an output; and
 - calculating a prediction using the output of the machine learning model for the medical image;
 - calculating a rate of change between the two or more predictions for the two or more medical images; and
 - creating a trained machine learning model by updating weights of the machine learning model to minimise a loss value, wherein the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change.
2. The method of claim 1, wherein the method further comprises determining a polynomial function of rate of change of the biomedical factor and calculating the expected rate of change using the polynomial function.
3. The method of claim 2, wherein calculating the expected rate of change using the polynomial function comprises applying the polynomial function to a mean value of the biomedical factor calculated by an average of the two or more predictions.
4. The method of claim 2 or 3, wherein determining the polynomial function comprises calculating a rate of change and a mean value for each pair of medical images in the patient study.
5. The method of any one of the preceding claims, wherein the loss value further comprises a change penalty value reflecting a difference between the two or more predictions, the change penalty value penalising an unexpected temporal trend of the biomedical factor related to the expected rate of change.

6. The method of any one of the preceding claims, wherein the loss value further comprises a correction penalty value reflecting a difference between the two or more predictions and two or more corresponding ground truth values, the correction penalty value penalising the two or more predictions from deviating from the two or more corresponding ground truth values.
7. The method of any one of the preceding claims, wherein the loss value further comprises an additional penalty value calculated by one or more of:
 - a difference between each of the two or more outputs of the machine learning model and a respective ground truth measurement; and
 - a mean square error between each of the two or more outputs and the respective ground truth measurement.
8. The method of claim 7, wherein the loss value is calculated by a weighted sum of the penalty value, the change penalty value, the correction penalty value, and the additional penalty value.
9. The method of any one of the preceding claims, wherein each of the medical images of the patient study has a timestamp and at least two of the two or more medical images have non-adjacent timestamps.
10. The method of any one of the preceding claims, wherein the method further comprises
 - calculating a further two or more predictions of the biomedical factor by performing for each of a further two or more medical images of the patient study:
 - applying the trained machine learning model to the medical image to calculate an output; and
 - calculating a prediction using the output;
 - calculating a further rate of change between the further two or more predictions; and
 - creating a further trained machine learning model by updating weights of the trained machine learning model to minimise a further loss value based of the further rate of change.

11. The method of any one of the preceding claims, wherein the method further comprises
- calculating another two or more predictions of the biomedical factor by performing for each of two or more medical images of another patient study different to the patient study:
 - applying the trained machine learning model or further trained machine learning model to the medical image to calculate an output; and
 - calculating a prediction using the output;
 - calculating another rate of change between the other two or more predictions; and
 - creating another trained machine learning model by updating weights of the trained machine learning model or further trained machine learning model to minimise another loss value based on the other rate of change.
12. The method of any one of the preceding claims, wherein the method further comprises applying the trained machine learning model to a medical image of a patient to aid in diagnosis of the patient based on the output of the trained machine learning model.
13. The method of any one of the preceding claims, wherein each of the medical images is one of a:
- magnetic resonance image;
 - positron emission tomography (PET) image; or
 - computed tomography (CT) image.
14. The method of any one of the preceding claims, wherein the biomedical factor is one of:
- medical image quantification factor;
 - a biological structure value; or
 - a biomarker.
15. The method of claim 14, wherein the medical image quantification factor is a Centiloid or standardised uptake value ratio scaling factor and the biological structure value is volume of a biological structure visible in the medical image.

16. The method of any one of claims 11 to 15, wherein each of the medical images is a PET image and the method further comprises performing PET quantification by applying the trained machine learning model to a PET image of a patient.

17. The method of claim 15 or 16, wherein
the medical image quantification factor is a Centiloid or standardised uptake value ratio scaling factor; and

the method further comprises applying the trained machine learning model to a medical image of a patient to aid in diagnosis of Alzheimer's disease based on the Centiloid or standardised uptake value ratio scaling factor that is an output of the trained machine learning model.

18. The method of any one of the preceding claims, further comprising generating one or more masks that correspond to corrections determined by the trained machine learning model.

19. Software that, when installed on a computer and executed by the computer, causes the computer to perform the method of any one of the preceding claims.

20. A system for training a machine learning model on longitudinal medical scan images of a patient study, the system comprises:

a processor configured to:

calculate two or more predictions of a biomedical factor by performing for each of two or more of the medical images:

applying the machine learning model to the medical image to calculate an output; and

calculating a prediction using the output of the machine learning model for the medical image;

calculate a rate of change between the two or more predictions for the two or more medical images; and

create a trained machine learning model by updating weights of the machine learning model to minimise a loss value, wherein the loss value comprises a penalty value that reflects a difference between the rate of change and an expected rate of change.

1/8

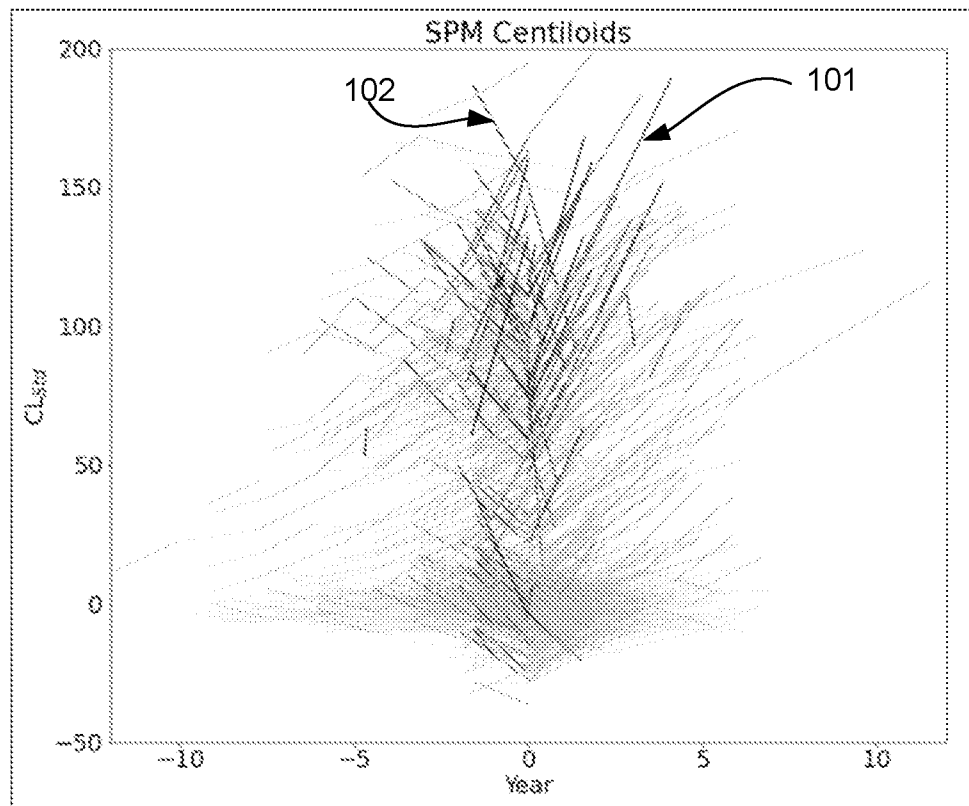


Fig. 1

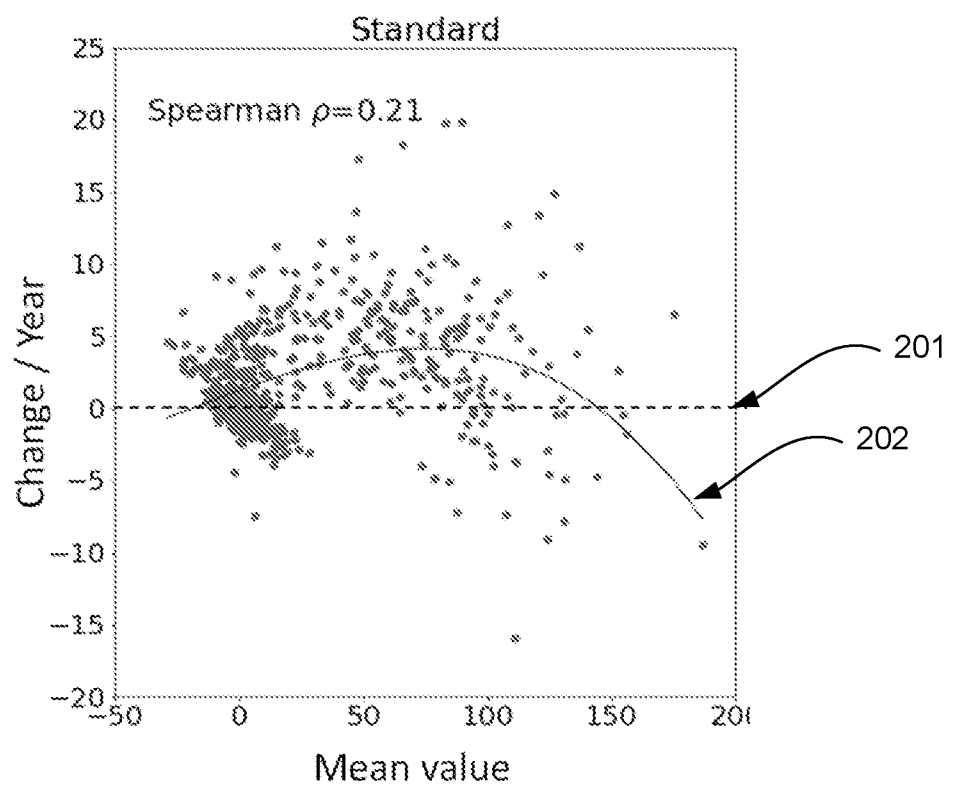


Fig. 2

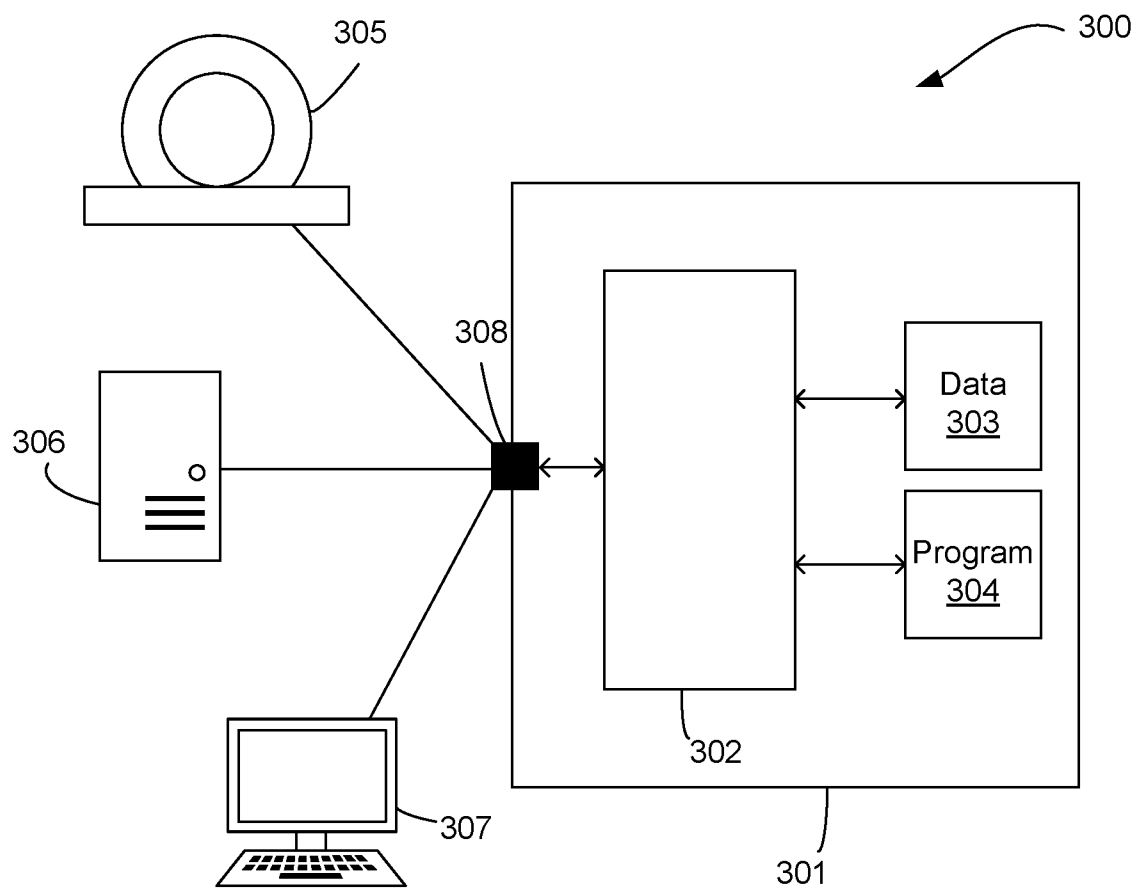


Fig. 3

3/8

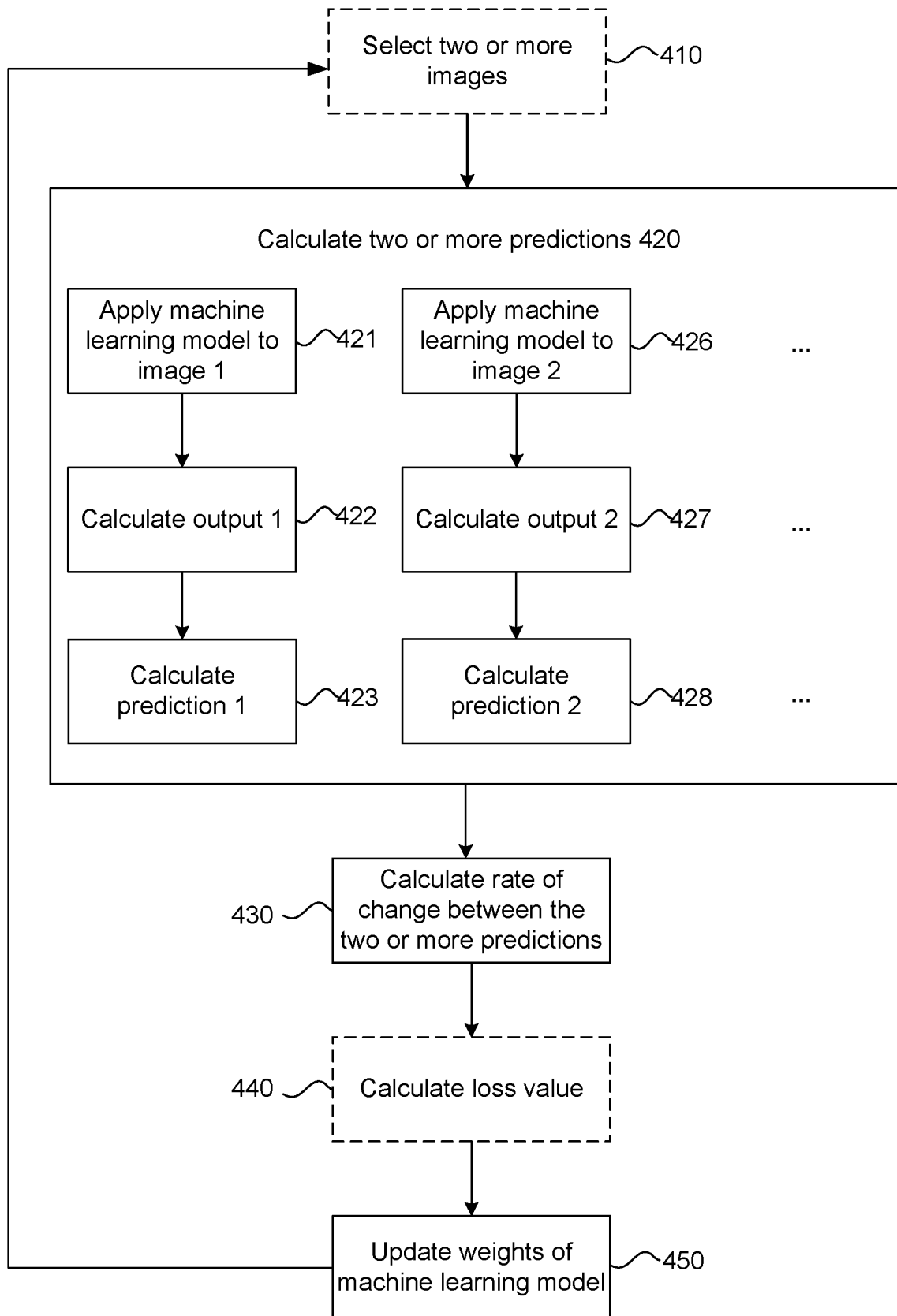


Fig. 4

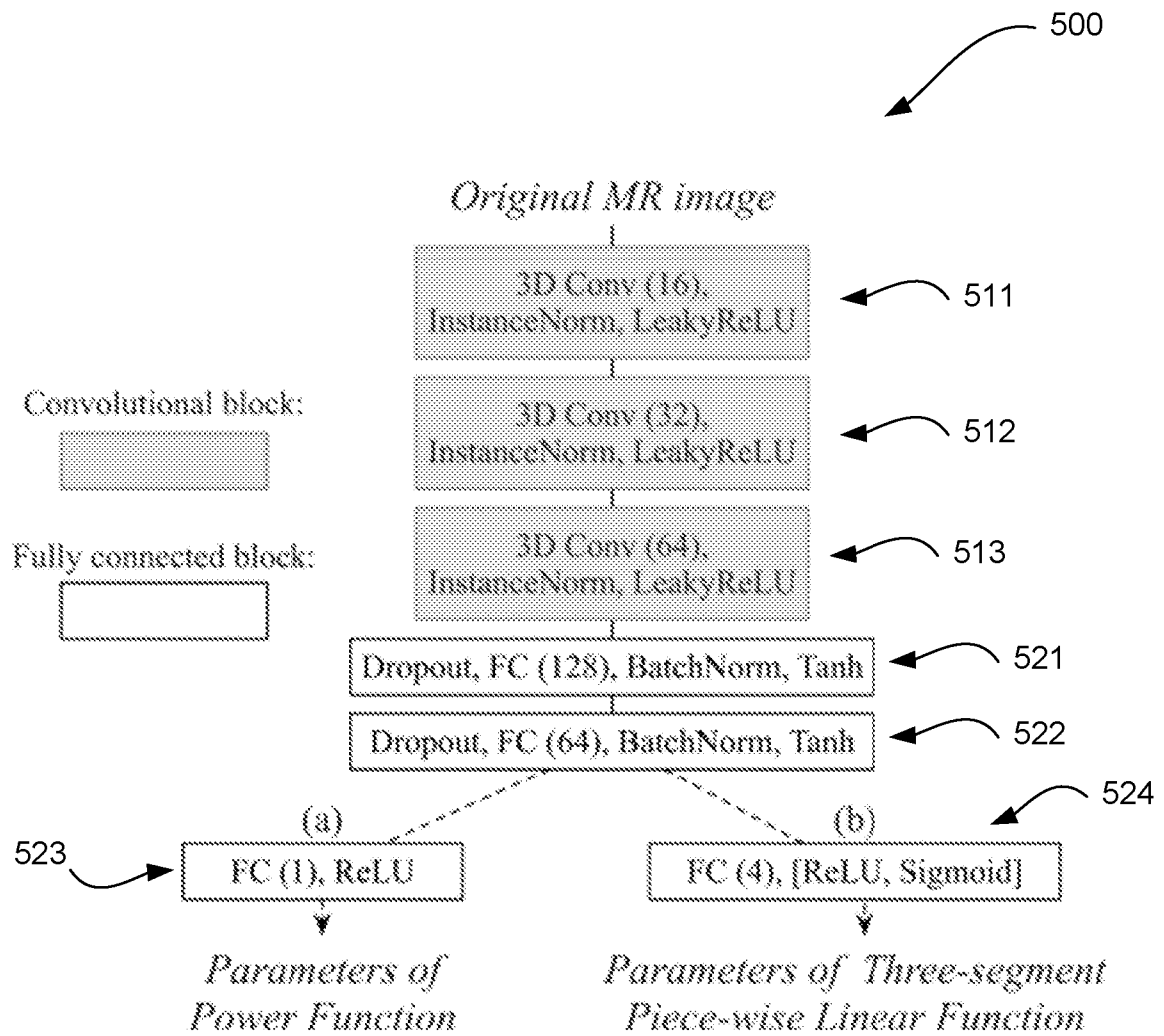


Fig. 5

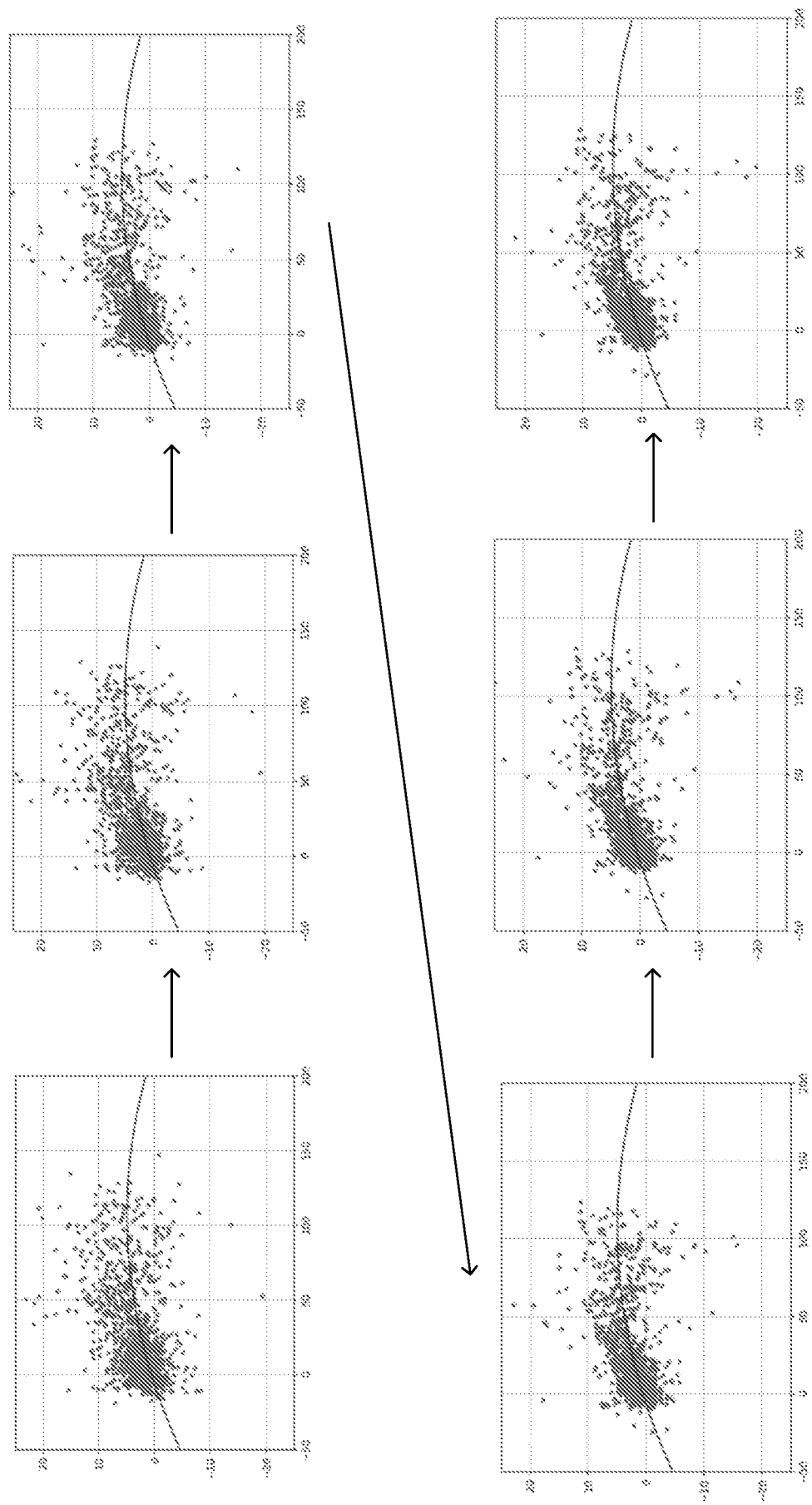


Fig. 6

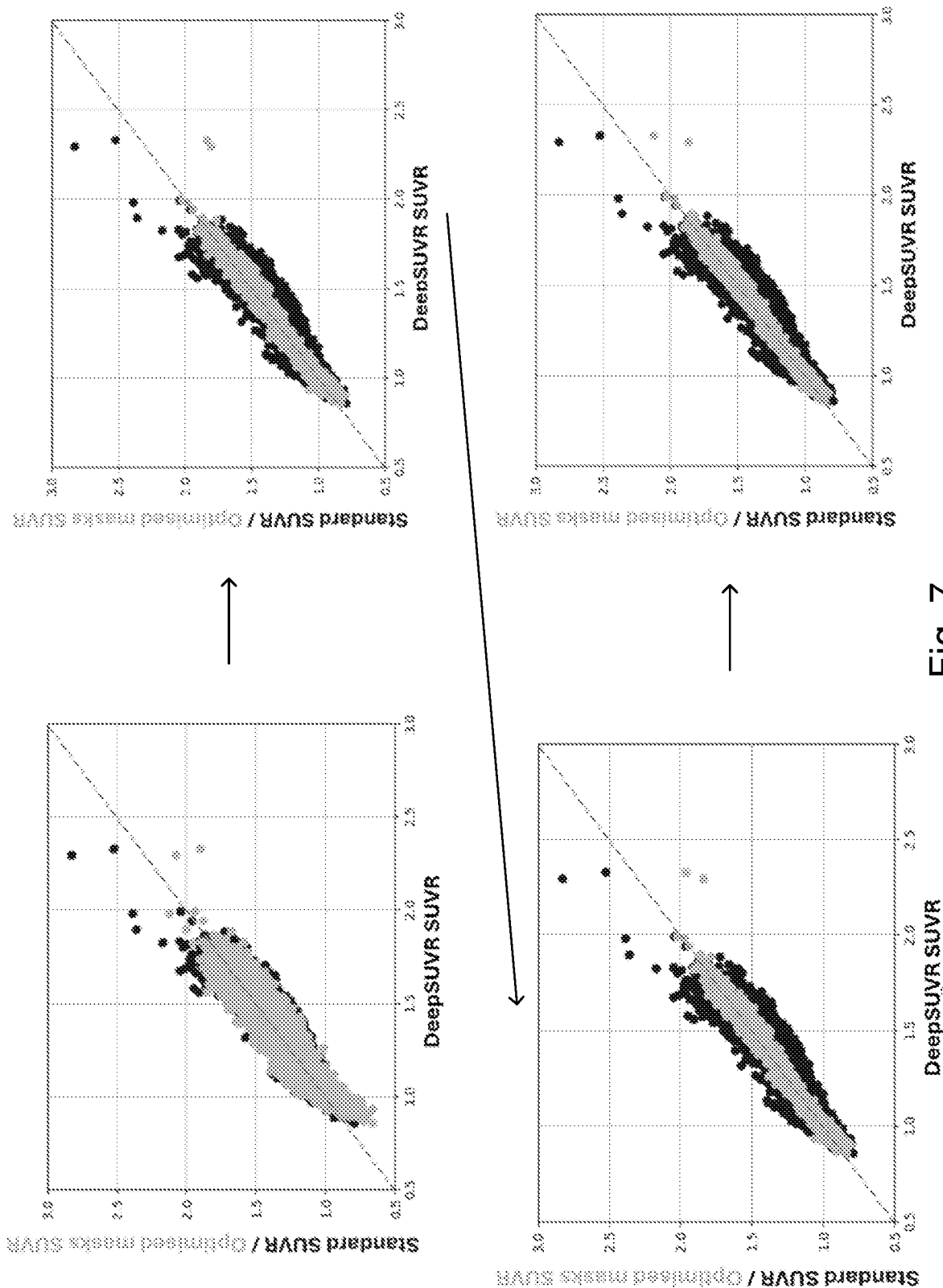


Fig. 7

7/8

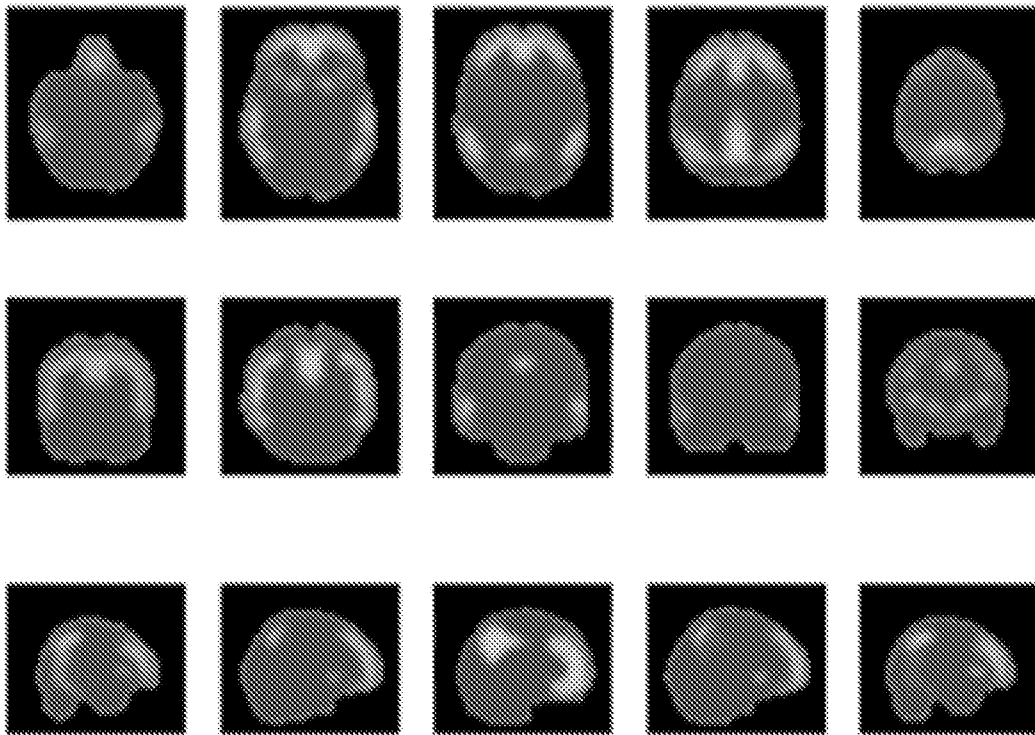


Fig. 8a

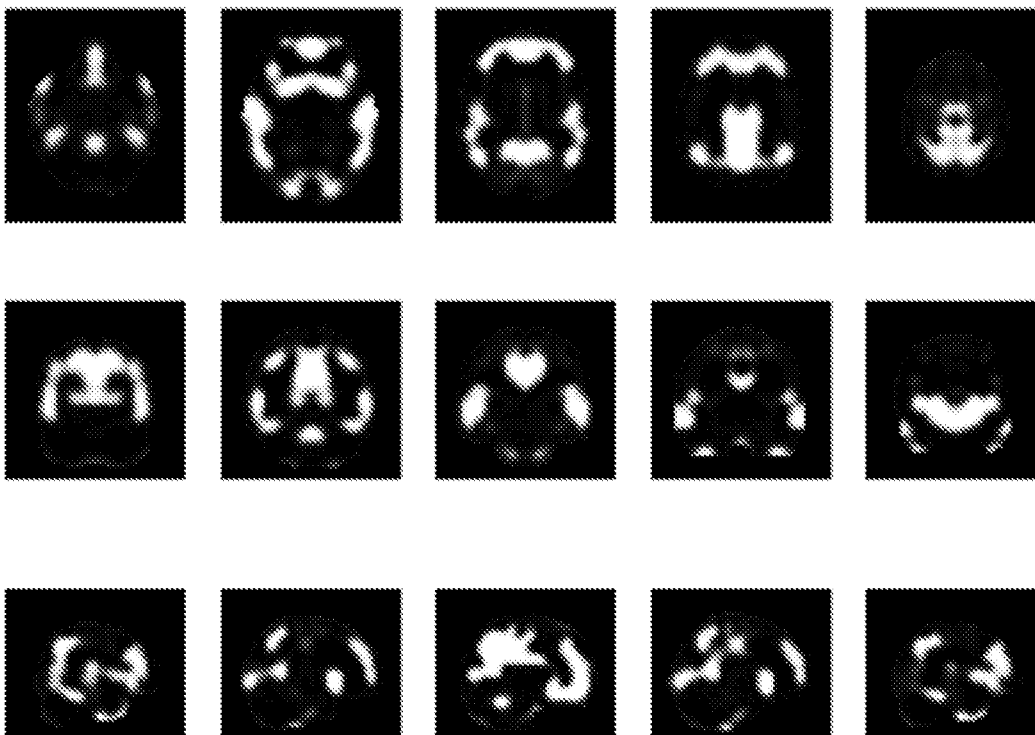


Fig. 8b

8/8

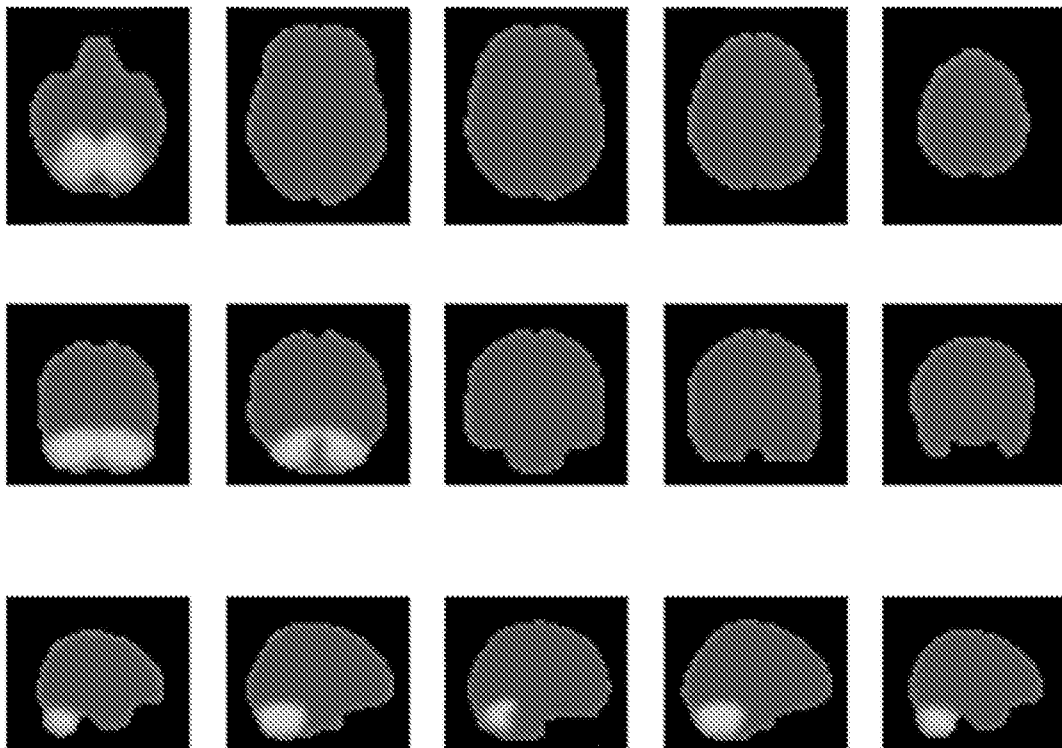


Fig. 9a



Fig. 9b

INTERNATIONAL SEARCH REPORT

International application No.

PCT/AU2024/051378

A. CLASSIFICATION OF SUBJECT MATTER

G06N 3/084 (2023.01) G06N 3/0464 (2023.01) G06N 20/00 (2019.01) G06T 7/00 (2017.01) G16H 30/40 (2018.01) G16H 50/20 (2018.01)

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)

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)

Databases: **PATENW** (all English language full-text databases): Keywords include: additional, adjust, alter, Alzheimer, average, biomarker, biomedical, brain, Centiloid, change, CNN, coefficient, compensate, convolutional, correct, corrections, correlation, CT, delta, dementia, difference, expected, explainable, factor, function, ground-truth, images, longitudinal, loss, machine-learning, mask, mean-square, medical, minimize, modify, MRI, multi, network-neural, optimize, over-correct, patient, Pearson, penalty, PET, plural, polynomial, prediction, quantification, rate-of-change, rate-of-growth, ratio, scans, score, second, several, standard, sum, SUVR, total, tracer, training, two, update, uptake, value, weight, and like terms. **Espacenet, Google, Google Patents, Google Scholar, The Lens**: Similar keywords as above. Also **DOCDB, DWPI, IP Australia** internal databases and **Google (including Scholar & Patents)** for applicant and inventor names.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	Documents are listed in the continuation of Box C	

☒ 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	"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
"D" document cited by the applicant in the international application	"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
"E" earlier application or patent but published on or after the international filing date	"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
"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)	"&" document member of the same patent family
"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
24 January 2025

Date of mailing of the international search report
24 January 2025

Name and mailing address of the ISA/AU

AUSTRALIAN PATENT OFFICE
PO BOX 200, WODEN ACT 2606, AUSTRALIA
Email address: pct@ipaustalia.gov.au

Authorised officer

Robert Foster
AUSTRALIAN PATENT OFFICE
(ISO 9001 Quality Certified Service)
Telephone No. +61 2 6222 3617

INTERNATIONAL SEARCH REPORT		International application No.
C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		PCT/AU2024/051378
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2023/102229 A1 (UNIVERSITY OF PITTSBURGH) 08 June 2023 Entire document, especially: FIG.'s 2, 5-7 & 14 and [0009], [0047], [0050]-[0053], [0057], [0059]-[0061], [0065]-[0067] & [0069]-[0071].	1-20
X	SCHALKAMP, A-K. et al., "Digital markers from smartwatch data relate to non-motor clinical examinations of Parkinson's disease", medRxiv, published September 2023 < URL: https://www.medrxiv.org/content/medrxiv/early/2023/09/12/2023.09.12.23295406.full.pdf > Entire document, especially: Figure 4 and pages 15 & 18-20.	1, 12, 14, 19-20
X	US 2023/0277118 A1 (SPARK NEURO INC) 07 September 2023 Entire document, especially: Abstract, FIG.'s 1-5 and [0005], [0012], [0019], [0039], [0046], [0071], [0111]-[0112], [0117], [0128], [0133] & [0173].	1, 20
X	WO 2023/164051 A1 (ARTERA INC) 31 August 2023 Entire document, especially: Abstract, FIG. 1 and [0058], [0065], [0121]-[0122], [0127], [0173], [0191] & [0198].	1, 20
A	SHISHEGAR, R. et al., "Using imputation to provide harmonized longitudinal measures of tton across AIBL and ADNI", Scientific reports 11.1, published 2021 < URL: https://www.nature.com/articles/s41598-021-02827-6.pdf > Entire document.	
A	WU, J., "Mining Associations between MRI Morphometry Measurements and Beta-Amyloid/tau Burden", Arizona State University, published 2022 < URL: https://keep.lib.asu.edu/system/files/c7/Wu_asu_0010E_22145.pdf > Entire document.	
A	BATTLE, M. R. et al., "Centiloid scaling for quantification of brain amyloid with [18F]flutemetamol using multiple processing methods", EJNMMI Research (2018) 8.107, published 2018, pages 1-11 < URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6281542/pdf/13550_2018_Article_456.pdf > Entire document.	
A	US 2021/0090694 A1 (TEMPUS LABS) 25 March 2021 Entire document.	
L	RONNEBERGER, O. et al., "U-Net: Convolutional Networks for Biomedical Image Segmentation", arXiv preprint arXiv:1505.04597, published 18 May 2015 < URL: https://arxiv.org/pdf/1505.04597 > Referred to by D1 at [0057]. See entire document, especially FIG. 3.	
A	LUCKETT, E. S. et al., "Association of Alzheimer's disease polygenic risk scores with amyloid accumulation in cognitively intact older adults", Alzheimer's Research & Therapy (2022) 14.138, published 2022 < URL: https://link.springer.com/content/pdf/10.1186/s13195-022-01079-4.pdf > Entire document.	
A	GIORGIO, J. et al., "A robust and interpretable machine learning approach using multimodal biological data to predict future pathological tau accumulation", Nature Communications (2022)13:1887 < URL: https://www.nature.com/articles/s41467-022-28795-7.pdf > Entire document.	
Form PCT/ISA/210 (fifth sheet) (July 2019)		

INTERNATIONAL SEARCH REPORT		International application No.
C (Continuation).	DOCUMENTS CONSIDERED TO BE RELEVANT	PCT/AU2024/051378
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P,A	HL, D.S. et al., "A Multimodal Approach Integrating Convolutional and Recurrent Neural Networks for Alzheimer's Disease Temporal Progression Prediction", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CPVR 2024), published June 2024 < URL: https://openaccess.thecvf.com/content/CVPR2024W/DEF-AI-MIA/papers/HL_A_Multimodal_Approach_Integrating_Convolutional_and_Recurrent_Neural_Networks_for_CVPRW_2024_paper.pdf > Entire document.	

Form PCT/ISA/210 (fifth sheet) (July 2019)

INTERNATIONAL SEARCH REPORT		International application No.	
Information on patent family members		PCT/AU2024/051378	
This Annex lists known patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.			
Patent Document/s Cited in Search Report		Patent Family Member/s	
Publication Number	Publication Date	Publication Number	Publication Date
WO 2023/102229 A1	08 June 2023	WO 2023102229 A1	08 Jun 2023
		CA 3241113 A1	08 Jun 2023
		EP 4441760 A1	09 Oct 2024
US 2023/0277118 A1	07 September 2023	US 2023277118 A1	07 Sep 2023
		WO 2023150266 A1	10 Aug 2023
WO 2023/164051 A1	31 August 2023	WO 2023164051 A1	31 Aug 2023
		EP 4483381 A1	01 Jan 2025
US 2021/0090694 A1	25 March 2021	US 2021090694 A1	25 Mar 2021
		US 11705226 B2	18 Jul 2023
		AU 2019359878 A1	27 May 2021
		AU 2019392537 A1	01 Jul 2021
		AU 2020274091 A1	09 Dec 2021
		AU 2020313915 A1	24 Feb 2022
		AU 2020326626 A1	10 Mar 2022
		AU 2020334901 A1	24 Mar 2022
		CA 3116712 A1	23 Apr 2020
		CA 3122070 A1	11 Jun 2020
		CA 3140365 A1	19 Nov 2020
		CA 3147100 A1	21 Jan 2021
		CA 3148023 A1	25 Feb 2021
		CA 3149381 A1	11 Feb 2021
		EP 3857555 A1	04 Aug 2021
		EP 3891755 A1	13 Oct 2021
		EP 3970152 A1	23 Mar 2022
		EP 3997243 A1	18 May 2022
		EP 4007522 A1	08 Jun 2022
		EP 4013866 A1	22 Jun 2022
		EP 4104175 A1	21 Dec 2022
		JP 2022532897 A	20 Jul 2022
		JP 2022543240 A	11 Oct 2022
Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001.			

Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001.
Form PCT/ISA/210 (Family Annex)(July 2019)

INTERNATIONAL SEARCH REPORT		International application No.	
Information on patent family members		PCT/AU2024/051378	
This Annex lists known patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.			
Patent Document/s Cited in Search Report		Patent Family Member/s	
Publication Number	Publication Date	Publication Number	Publication Date
		JP 2022544604 A	19 Oct 2022
		US 10395772 B1	27 Aug 2019
		US 2020126663 A1	23 Apr 2020
		US 10902952 B2	26 Jan 2021
		US 2020176098 A1	04 Jun 2020
		US 10957433 B2	23 Mar 2021
		US 2021012882 A1	14 Jan 2021
		US 10978196 B2	13 Apr 2021
		US 2021057042 A1	25 Feb 2021
		US 11367508 B2	21 Jun 2022
		US 2020365268 A1	19 Nov 2020
		US 11527323 B2	13 Dec 2022
		US 2021151192 A1	20 May 2021
		US 11532397 B2	20 Dec 2022
		US 2021233664 A1	29 Jul 2021
		US 11640859 B2	02 May 2023
		US 2022181029 A1	09 Jun 2022
		US 11651442 B2	16 May 2023
		US 2021350904 A1	11 Nov 2021
		US 11682481 B2	20 Jun 2023
		US 2023223121 A1	13 Jul 2023
		US 12112839 B2	08 Oct 2024
		US 2020258601 A1	13 Aug 2020
		US 2020365232 A1	19 Nov 2020
		US 2021142904 A1	13 May 2021
		US 2021210184 A1	08 Jul 2021
		US 2022351805 A1	03 Nov 2022
		US 2023187070 A1	15 Jun 2023
		US 2023245788 A1	03 Aug 2023
		US 2023253070 A1	10 Aug 2023
		US 2023267553 A1	24 Aug 2023
		US 2023386643 A1	30 Nov 2023
		US 2024312581 A1	19 Sep 2024
		WO 2020081795 A1	23 Apr 2020
		WO 2020117869 A1	11 Jun 2020
		WO 2020232033 A1	19 Nov 2020
Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001.			

INTERNATIONAL SEARCH REPORT Information on patent family members		International application No. PCT/AU2024/051378	
This Annex lists known patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.			
Patent Document/s Cited in Search Report		Patent Family Member/s	
Publication Number	Publication Date	Publication Number	Publication Date
		WO 2021011507 A1	21 Jan 2021
		WO 2021026097 A1	11 Feb 2021
		WO 2021034712 A1	25 Feb 2021
		WO 2021163233 A1	19 Aug 2021
		WO 2023224709 A1	23 Nov 2023
End of Annex			
Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001. Form PCT/ISA/210 (Family Annex)(July 2019)			