

Method and System for Precision Face Lookup and Identification using Multilayer Ensembles

FIELD OF INVENTION

5

[0001] The present disclosure relates broadly, but not exclusively, to methods and systems for precision face lookup and identification using multilayer ensembles.

BACKGROUND

10

[0002] Face matching is widely used to validate and authenticate people, for example such as turning on an electronic device, opening a door, and other similar actions by presenting a face in front of a camera for authentication. Face based authentication may be used in conjunction with other existing authentication mechanisms such as user name/password, fingerprints etc. A basic flow for such authentication methods is to evaluate a face that is presented for the authentication against a known reference face.

15

[0003] Other applications that are not just for authentication may also benefit from face matching, in which lookup and identification of a face are required, possibly followed by authentication thereafter. Some of these applications may be extremely sensitive in nature. For example, they may be related to payments, security or safety issues related to criminal incidents. It is imperative that a system that has high accuracy and high precision in lookup and identification of a face is used for such applications.

20

[0004] A need therefore exists to provide methods and systems that seek to overcome or at least minimize the above mentioned challenges.

25

SUMMARY

[0005] According to a first aspect of the present disclosure, there is provided a method for identifying a target face depicted in a target face image from a plurality of face images, each of the plurality of face images depicting a face, comprising: identifying a plurality of matching face images from the plurality of face images based on the target face using a plurality of models running independently of and in parallel with one another, each of the plurality of models being configured to compare the target face with each face depicted in the plurality of face images, each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face; and determining, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being configured to compare each of the plurality of matching face images with one another for the determination, wherein the ensemble model determines that a face depicted in a matching face image is an exact match with the target face if the matching face image is one that is identified by all of the plurality of models.

[0006] According to a second aspect of the present disclosure, there is provided a system for identifying a target face depicted in a target face image from a plurality of face images, each of the plurality of face images depicting a face, the system comprising: at least one processor; and at least one memory including computer program code; the at least one memory and the computer program code configured to, with the at least one processor, cause the system at least to: identify a plurality of matching face images from the plurality of face images based on the target face using a plurality of models running independently of and in parallel with one another, each of the plurality of models being configured to compare the target face with each face depicted in the plurality of face images, each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face; and determine, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being configured to compare each of the plurality of matching face images with one another for the determination, wherein the ensemble model determines that a face depicted in a matching face image is an exact match with the target face if the matching face image is one that is identified by all of the plurality of models.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] Embodiments and implementations are provided by way of example only, and will be better understood and readily apparent to one of ordinary skill in the art from the following written description, read in conjunction with the drawings, in which:

[0008] Fig. 1 illustrates a system for precision face lookup and identification using multi-layer ensembles according to various embodiments of the present disclosure.

[0009] Fig. 2 is a schematic diagram of a face recognition server, according to various embodiments of the present disclosure.

[0010] Fig. 3 is an overview of a process for precision face lookup and identification using multilayer ensembles, according to various embodiments.

[0011] Fig. 4 depicts an example illustration of a false positive during a face matching process.

[0012] Fig. 5 illustrates an example flow diagram of how a target face may be identified from a plurality of face images using a one-to-one (1-1) match implementation according to various embodiments.

[0013] Fig. 6 illustrates an example flow diagram of how a target face may be identified from a plurality of face images using a combined one-to-many (1-N) match and one-to-one (1-1) match implementation according to various embodiments.

[0014] Fig. 7 illustrates an example flow diagram of how retraining of an ensemble model may be implemented according to various embodiments.

[0015] Figs. 8A and 8B form a schematic block diagram of a general purpose computer system upon which the transaction processing server of Fig. 1 can be practiced.

[0016] Fig. 8C is a schematic block diagram of a general purpose computer system upon which the face recognition server of Fig. 2 can be practiced.

[0017] Fig. 8D is a schematic block diagram of a general purpose computer system upon which a combined transaction processing server and face recognition server of Fig. 1 can be practiced.

5 **[0018]** Fig. 9 shows an example of a computing device to realize the transaction processing server shown in Fig. 1.

[0019] Fig. 10 shows an example of a computing device to realize the face recognition server shown in Fig. 1.

10

[0020] Fig. 11 shows an example of a computing device to realize a combined transaction processing and face recognition server shown in Fig. 1.

15

[0021] Skilled artisans will appreciate that elements in the figures are illustrated for simplicity and clarity and have not necessarily been depicted to scale. For example, the dimensions of some of the elements in the illustrations, block diagrams or flowcharts may be exaggerated in respect to other elements to help to improve understanding of the present embodiments.

20

DETAILED DESCRIPTION

[0022] Embodiments will be described, by way of example only, with reference to the drawings. Like reference numerals and characters in the drawings refer to like elements or equivalents.

25

[0023] Some portions of the description which follows are explicitly or implicitly presented in terms of algorithms and functional or symbolic representations of operations on data within a computer memory. These algorithmic descriptions and functional or symbolic representations are the means used by those skilled in the data processing arts to convey most effectively the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities, such as electrical, magnetic or optical signals capable of being stored, transferred, combined, compared, and otherwise manipulated.

30

[0024] Unless specifically stated otherwise, and as apparent from the following, it will be appreciated that throughout the present specification, discussions utilizing terms such as “monitoring”, “utilizing”, “retrieving”, “providing”, “generating”, “quantifying”, “calculating”, “outputting”, “optimising”, “rebuilding”, “storing”, “mapping”, “checking”, “identifying”, “collecting”, “searching”, “conducting”, “cross-checking”, “aggregating”, “determining”, “regenerating”, “updating”, “comparing”, “adjusting”, “compiling”, “performing”, “obtaining”, “predicting” or the like, refer to the action and processes of a computer system, or similar electronic device, that manipulates and transforms data represented as physical quantities within the computer system into other data similarly represented as physical quantities within the computer system or other information storage, transmission or display devices.

[0025] In addition, the present specification also implicitly discloses a computer program, in that it would be apparent to the person skilled in the art that the individual steps of the method described herein may be put into effect by computer code. The computer program is not intended to be limited to any particular programming language and implementation thereof. It will be appreciated that a variety of programming languages and coding thereof may be used to implement the teachings of the disclosure contained herein. Moreover, the computer program is not intended to be limited to any particular control flow. There are many other variants of the computer program, which can use different control flows without departing from the scope of the specification.

[0026] Furthermore, one or more of the steps of the computer program may be performed in parallel rather than sequentially. Such a computer program may be stored on any computer readable medium. The computer readable medium may include storage devices such as magnetic or optical disks, memory chips, or other storage devices suitable for interfacing with a computer. The computer readable medium may also include a hard-wired medium such as exemplified in the Internet system, or wireless medium such as exemplified in the GSM mobile telephone system. The computer program when loaded and executed on such a computer effectively results in an apparatus that implements the steps of the preferred method.

[0027] Face recognition refers to a process in which a human face from, for example, a digital image, a video frame, or other similar representations, is matched against a database of faces. This is typically employed to authenticate users through ID verification services, and works by pinpointing and measuring facial features from a given image. In some

techniques, each of the input face and the database of faces may be vectorized, in which a set of unique key features (e.g. facial feature vectors wherein each vector correspond to a particular facial feature of an associated face) is extracted for each face. By comparing the extracted feature vectors of the input face with corresponding extracted feature vectors of each of the faces in the database, the input face may then be identified. These vectors may be extracted by deep neural network models or models employing other types of algorithms.

[0028] In statistics and machine learning, ensemble methods refer to the use of multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. In the present disclosure, ensembles may similarly be used for face recognition, in which multiple algorithms for face lookup and identification are used to obtain better face recognition performance than could be obtained from any of the constituent algorithms alone. Empirically, ensembles tend to yield better results when there is a significant diversity among the models (e.g each model comprising a different algorithm) used. It is therefore beneficial to promote diversity among the models and algorithms combined for use in ensemble methods. For example, a plurality of models each comprising a different algorithm may be configured to run independently and in parallel with one another for face lookup and identification of an input face from a plurality of faces.

[0029] A process for lookup and identification of a face could be summarized as follows: given an input face, search for similar faces from a pool of 50 to 100 million images in real time (e.g. lookup phase), and evaluate the candidate faces against the query image for high precision/accuracy match (e.g. identification phase). False positives, in which a candidate face is mistakenly identified as matching with the input face, may occur. The stakes for false positives are high. In an example payment scenario, someone else's wallet may be charged for a purchase of a product or service. In a criminal incident scenario, innocent people may be prosecuted due to a false positive face match.

[0030] Due to prevalence of false positives even among commercially available face recognition services (typically 1%-5% of obtained results), human review may be required as a follow up and safeguard measure. For example, Fig. 4 depicts an example illustration in which face 402 and face 404 are considered by a commercially available face recognition service as a high confidence match of each other. However, in this case, it is easy for

a human to realize that these are actually faces of different people. Such mistakes would mean that some commercially available vendor solutions for face recognition are unusable for payment authentication or any other sensitive applications.

5 **[0031]** In the present disclosure, a plurality of models running independently and in parallel with one another are utilized to greatly improve quality of lookup and identification process for face recognition. Firstly, given a target face depicted in, for example, an target face image, multiple parallel independent algorithms are used for both lookup (one-to-many or 1:n) and matching (one-to-one or 1:1) algorithms to identify faces that match the
10 target face from a plurality of face images, wherein each of the plurality of face images depict a face. Lookup itself is a two step process, where additional parallel algorithms may be added for both face vectorization, and nearest neighbor or cluster search. The faces that were determined by these algorithms to be similar to or matching with the target face may be termed as matching faces. Secondly, a followup 1:1 face matching is performed
15 on each matching face.

[0032] A user may be any suitable type of entity, which may include a person, a consumer looking to purchase a good or service via a transaction processing server, a seller looking to sell a good or service via the transaction processing server, a motorcycle driver or pillion
20 rider in a case of the user looking to book or provide a motorcycle ride via the transaction processing server, a car driver or passenger in a case of the user looking to book or provide a car ride via the transaction processing server, and other similar entity. A user who is registered to the transaction processing or face recognition server will be called a registered user. A user who is not registered to the transaction processing server or face
25 recognition server will be called a non-registered user. The term user will be used to collectively refer to both registered and non-registered users. A user may interchangeably be referred to as a requestor (e.g. a person who requests for a good or service) or a provider (e.g. a person who provides the requested good or service to the requestor).

30 **[0033]** A face recognition server is a server that hosts software application programs for performing face recognition. The face recognition server may be implemented as shown in schematic diagram 300 of Fig. 3 for identifying a face of the user.

[0034] The transaction processing server is a server that hosts software application programs for processing payment transactions for, for example, purchasing of a good or service by a user. The transaction processing server may also be configured for processing travel co-ordination requests between a requestor and a provider. The transaction processing server communicates with any other servers (e.g., a face recognition server) concerning processing payment transactions or travel co-ordination requests. The transaction processing server communicates with a face recognition server to facilitate user authentication for purchase of a good or service, or for a ride associated with the travel co-ordination request. The transaction processing server may use a variety of different protocols and procedures in order to process the payment and/or travel co-ordination requests.

[0035] Transactions that may be performed via a transaction processing server include product or service purchases, credit purchases, debit transactions, fund transfers, account withdrawals, etc. Transaction processing servers may be configured to process transactions via cash-substitutes, which may include payment cards, letters of credit, checks, payment accounts, etc.

[0036] The transaction processing server is usually managed by a service provider that may be an entity (e.g. a company or organization) which operates to process transaction requests and/or travel co-ordination requests e.g. pair a provider of a travel co-ordination request to a requestor of the travel co-ordination request. The travel co-ordination server may include one or more computing devices that are used for processing transaction requests and/or travel co-ordination requests.

[0037] A transaction account is an account of a user who is registered at a transaction processing server. The user can be a customer, a hail provider (e.g., a driver), or any third parties (e.g., a courier) who want to use the transaction processing server. In certain circumstances, the transaction account is not required to use the face recognition server. A transaction account includes details (e.g., name, address, vehicle, face image, etc.) of a user. The transaction processing server manages the transaction accounts of users and the interactions between users and other external servers.

[0038] Fig. 1 illustrates a block diagram of a system 100 for precision face lookup and identification using multilayer ensembles. Further, the system 100 enables a payment

transaction for a good or service, and/or a request for a ride between a requestor and a provider.

5 **[0039]** The system 100 comprises a requestor device 102, a provider device 104, an acquirer server 106, a transaction processing server 108, an issuer server 110, a face recognition server 140 and a reference image database 150.

10 **[0040]** The requestor device 102 is in communication with a provider device 104 via a connection 112. The connection 112 may be wireless (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet). The requestor device 102 is also in communication with the face recognition server 140 via a connection 121. The connection 121 may be a network (e.g., the Internet). The requestor device 102 may also be connected to a cloud that facilitates the system 100 for precision face lookup and identification using multilayer ensembles. For example, the requestor device 102 can send a signal or data to the cloud directly via a wireless connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

20 **[0041]** The provider device 104 is in communication with the requestor device 102 as described above, usually via the transaction processing server 108. The provider device 104 is, in turn, in communication with an acquirer server 106 via a connection 114. The provider device 104 is also in communication with the face recognition server 140 via a connection 123. The connections 114 and 123 may be a network (e.g., the Internet). The provider device 104 may also be connected to a cloud that facilitates the system 100 for precision face lookup and identification using multilayer ensembles. For example, the provider device 104 can send a signal or data to the cloud directly via a wireless connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

30 **[0042]** The acquirer server 106, in turn, is in communication with the transaction processing server 108 via a connection 116. The transaction processing server 108, in turn, is in communication with an issuer server 110 via a connection 118. The connections 116 and 118 may be a network (e.g., the Internet).

35 **[0043]** The transaction processing server 108 is further in communication with the face recognition server 140 via a connection 120. The connection 120 may be over a network (e.g., a local area network, a wide area network, the Internet, etc.). In one arrangement,

the transaction processing server 108 and the face recognition 140 are combined and the connection 120 may be an interconnected bus.

5 [0044] The face recognition server 140, in turn, is in communication with the reference image database 150 via respective connection 122. The connection 122 may be a network (e.g., the Internet). The face recognition server 140 may also be connected to a cloud that facilitates the system 100 for precision face lookup and identification using multilayer ensembles. For example, the face recognition server 140 can send a signal or data to the cloud directly via a wireless connection (e.g., via NFC communication, Bluetooth, etc.) or
10 over a network (e.g., the Internet).

[0045] The reference image database 150 comprises a plurality of images, wherein each image depicts a face of, for example, a person (e.g. a requestor or a provider) utilizing the transaction processing server 108. The reference image database may be combined with
15 the face recognition server 140. In an example, the reference image database 150 may be a database managed by an external entity and the face recognition server 140 is a server that, based on a face depicted in a target image, determines, from the plurality of images in the reference image database, an image that depicts a same face as that of the target image. The target image may be an image provided by the requestor or provider
20 via the requestor device 102 or provider device 104 respectively for authentication purposes before a transaction can be processed by the transaction processing server 108. Alternatively, a module such as a reference image module may store the plurality of images instead of the reference image database 150, wherein the reference image module may be integrated as part of the face recognition server 140 or external from the face
25 recognition server 140.

[0046] In the illustrative embodiment, each of the devices 102, 104, and the servers 106, 108, 110, 140, and 150 provides an interface to enable communication with other connected devices 102, 104, 142 and/or servers 106, 108, 110, 140, and 150. Such communication is facilitated by an application programming interface ("API"). Such APIs may be
30 part of a user interface that may include graphical user interfaces (GUIs), Web-based interfaces, programmatic interfaces such as application programming interfaces (APIs) and/or sets of remote procedure calls (RPCs) corresponding to interface elements, messaging interfaces in which the interface elements correspond to messages of a communication protocol, and/or suitable combinations thereof. For example, it is possible for at
35

least one of the requestor device 102 and the provider device 104 to send an image depicting a user's face in response to an enquiry shown on the GUI running on the respective API.

5 **[0047]** Use of the term 'server' herein can mean a single computing device or a plurality of interconnected computing devices which operate together to perform a particular function. That is, the server may be contained within a single hardware unit or be distributed among several or many different hardware units.

10 **[0048]** The face recognition server 140 is associated with an entity (e.g. a company or organization or moderator of the service). In one arrangement, the face recognition server 140 is owned and operated by the entity operating the transaction processing server 108. In such an arrangement, the face recognition server 140 may be implemented as a part (e.g., a computer program module, a computing device, etc.) of the transaction processing
15 server 108.

[0049] The transaction processing server 108 may also be configured to manage the registration of users. A registered user has a transaction account (see the discussion above) which includes details of the user. The registration step is called on-boarding. A user may
20 use either the requestor device 102 or the provider device 104 to perform on-boarding to the transaction processing server 108.

[0050] It may not be necessary to have a transaction account at the transaction processing server 108 to access the functionalities of the transaction processing server 108.
25 However, there are functions that are available to a registered user. These additional functions will be discussed below.

[0051] The on-boarding process for a user is performed by the user through one of the requestor device 102 or the provider device 104. In one arrangement, the user downloads
30 an app (which includes the API to interact with the transaction processing server 108) to the requestor device 102 or the provider device 104. In another arrangement, the user accesses a website (which includes the API to interact with the transaction processing server 108) on the requestor device 102 or the provider device 104. The user is then able to interact with the face recognition server 140. The user may be a requestor or a provider
35 associated with the requestor device 102 or the provider device 104, respectively.

[0052] Details of the registration include, for example, name of the user, address of the user, emergency contact, blood type or other healthcare information, next-of-kin contact, permissions to retrieve data and information from the requestor device 102 and/or the provider device 104 for face recognition purposes, such as permission to use a camera of the requestor device 102 and/or the provider device 104 to take a picture of the requestor's or provider's face, wherein the picture may be used by the face recognition server 140 as a target image for face recognition purposes. Alternatively, another mobile device may be selected instead of the requestor device 102 and/or the provider device 104 for retrieving the target image. Once on-boarded, the user would have a transaction account that stores all the details.

[0053] The requestor device 102 is associated with a customer (or requestor) who is a party to a travel request that occurs between the requestor device 102 and the provider device 104. The requestor device 102 may be a computing device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like.

[0054] The requestor device 102 includes transaction credentials (e.g., a payment account) of a requestor to enable the requestor device 102 to be a party to a payment transaction. If the requestor has a transaction account, the transaction account may also be included (i.e., stored) in the requestor device 102. For example, a mobile device (which is a requestor device 102) may have the transaction account of the customer stored in the mobile device.

[0055] In one example arrangement, the requestor device 102 is a computing device in a watch or similar wearable and is fitted with a wireless communications interface (e.g., a NFC interface). The requestor device 102 can then electronically communicate with the provider device 104 regarding a transaction request. The customer uses the watch or similar wearable to make request regarding the transaction request by pressing a button on the watch or wearable.

[0056] The provider device 104 is associated with a provider who is also a party to the transaction request that occurs between the requestor device 102 and the provider device

104. The provider device 104 may be a computing device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like.

5 **[0057]** Hereinafter, the term “provider” refers to a service provider and any third party associated with providing a good or service for purchase, or a travel or ride or delivery service via the provider device 104. Therefore, the transaction account of a provider refers to both the transaction account of a provider and the transaction account of a third party (e.g., a travel co-ordinator or merchant) associated with the provider.

10

[0058] If the provider has a transaction account, the transaction account may also be included (i.e., stored) in the provider device 104. For example, a mobile device (which is a provider device 104) may have the transaction account of the provider stored in the mobile device.

15

[0059] In one example arrangement, the provider device 104 is a computing device in a watch or similar wearable and is fitted with a wireless communications interface (e.g., a NFC interface). The provider device 104 can then electronically communicate with the requestor to make request regarding the transaction request by pressing a button on the watch or wearable.

20

[0060] The acquirer server 106 is associated with an acquirer who may be an entity (e.g. a company or organization) which issues (e.g. establishes, manages, administers) a payment account (e.g. a financial bank account) of a merchant. Examples of the acquirer include a bank and/or other financial institution. As discussed above, the acquirer server 106 may include one or more computing devices that are used to establish communication with another server (e.g., the transaction processing server 108) by exchanging messages with and/or passing information to the other server. The acquirer server 106 forwards the payment transaction relating to a transaction request to the transaction processing server 108.

25

30

[0061] The transaction processing server 108 is configured to process processes relating to a transaction account by, for example, forwarding data and information associated with the transaction to the other servers in the system 100 such as the face recognition server 140. In an example, the transaction processing server 108 may, instead of the requestor

35

device 102 or the provider device 104, transmit a target image to the face recognition server 140 for authentication via face recognition before a transaction can be processed by the transaction processing server 108. Further, the transaction processing server 108 may provide data and information associated with the target image and a plurality of images that are used for the face recognition process of face recognition server 140. The transaction may then be authenticated based on an outcome of the face recognition process e.g. when the target face depicted in the target image is identified as one of the faces depicted in the plurality of images, and data and information associated with the identified face corresponds with data and information associated with the user of the requestor device 102 or the provider device 104.

[0062] The issuer server 110 is associated with an issuer and may include one or more computing devices that are used to perform a payment transaction. The issuer may be an entity (e.g. a company or organization) which issues (e.g. establishes, manages, administers) a transaction credential or a payment account (e.g. a financial bank account) associated with the owner of the requestor device 102. As discussed above, the issuer server 110 may include one or more computing devices that are used to establish communication with another server (e.g., the transaction processing server 108) by exchanging messages with and/or passing information to the other server.

[0063] The reference image database 150 is a database or server associated with an entity (e.g. a company or organization) which manages (e.g. establishes, administers) data relating to a plurality of users that are registered with a transaction account with the transaction processing server 108. In one arrangement, the data comprises a plurality of face images wherein each image depicts a face belonging one of the plurality of users. The plurality of face images is used by the face recognition server 140 for verifying whether a face depicted in a target image is one belonging to a registered user of the transaction processing server 108.

[0064] Fig. 2 illustrates a schematic diagram of the face recognition server 140 according to various embodiments. The face recognition server 140 may comprise a data module 260 configured to receive data and information from the requestor device 102, provider device 104, transaction processing server 108, a cloud and other sources of information to facilitate the precision face lookup and identification using multilayer ensembles by the

face recognition server 140. For example, the data module 260 may be configured to receive a target image depicting a target face from the requestor device 102, the provider device 104, transaction processing server 108 or other sources of information. The data module 260 may also be configured to receive a plurality of face images each depicting a face from the transaction processing server 108, the reference image database 150 of other sources of information. The data module 260 may be further configured to send an image such as an output image obtained after a face recognition process to the transaction processing server 108, a database or other sources of information.

[0065] The face recognition server 140 may comprise a plurality of lookup models 262 running independently of and in parallel with one another, each of the plurality of lookup models being configured to compare the target face with each of the plurality of face images for determining which of the plurality of face images depict a face that is similar to the target face. Each of the plurality of lookup models may comprise a different face vectorization algorithm for vectorizing the target face, wherein determining which of the plurality of face images depict a face that is similar to the target face further comprises vectorizing the target face and comparing the vectorized target face with each face of the plurality of face images by each of the plurality of lookup models, wherein each face depicted in the plurality of face images is vectorized.

[0066] The face images that are determined to depict a face that is similar to the target face by each of the plurality of lookup models 262 may then be consolidated by a collection module 263 and provided as input into a plurality of models 264. The plurality of models 264 are configured to run independently of and in parallel with one another, wherein each of the plurality of models 264 is configured to compare the target face with each face depicted in the plurality of determined face images to identify a plurality of matching face images. Each of the plurality of matching face images is being identified by at least one of the plurality of models 264 as depicting a face that matches with the target face. Each of the plurality of models 264 may comprise a different face matching algorithm, wherein identifying matching face images from the plurality of face images further comprises comparing, by each of the different face matching algorithm, the target face with each face depicted in the plurality of face images.

[0067] The plurality of matching face images identified by the plurality of models may then be further processed by an ensemble module 266 which is configured to determine which

of the plurality of matching face images depict a face that is an exact match with the target face. The ensemble model 266 is configured to compare each of the plurality of matching face images with one another for the determination. For example, the ensemble model may determine that a face depicted in a matching face image is an exact match with the target face if the matching face image is one that is identified by all of the plurality of models 264. Further, weights may be assigned to each of the plurality of models 264, wherein the determination by the ensemble module 266 is based on the assigned weights. Based on the determination, the ensemble module 266 may then generate an output image, the output face image being one of the plurality of matching face images that is determined by the ensemble module 266 as depicting a face that is an exact match with the target face. Training data may also be generated by a retraining module 268 based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image. Weights may then be assigned to each of the plurality of models based on the training data. The retraining module 268 may also be configured to retrain the ensemble model based on the training data by reinforcement learning. Based on the retraining, the assignment of weights to the plurality of models 264 can be optimized. In an arrangement, the retraining module 268 may be configured to receive human input as to whether, for example, a matching face image is a false positive, so that retraining data can capture this information for optimizing the face recognition process.

[0068] Each of the data module 260, lookup models 262, collection module 263, models 264, ensemble module 266 and retraining module 268 may further be in communication with a processing module (not shown) of the face recognition server 140, for example for coordination of respective tasks and functions during the face recognition process. The data module 260 may be further configured to communicate with and store data and information for each of the processing module, lookup models 262, collection module 263, models 264, ensemble module 266 and retraining module 268. Alternatively, all the tasks and functions required for precision face lookup and identification using multilayer ensembles may be performed by a single processor of the face recognition server 140.

[0069] Fig. 3 depicts a schematic overview of a system 300 for precision face lookup and identification using multilayer ensembles, according to various embodiments. The system 300 comprises a plurality of models 312 running independently and in parallel with one another, wherein each of the plurality of models 312 are configured to compare a target face 302 with each face depicted in a plurality of face images. The system 300 may be

implemented based on or as the face recognition server 200 of Figure 2 for precision face lookup and identification using multilayer ensembles. For example, the system 300 may also be implemented as a mobile device, a backend system, or other similar implementations that facilitate face lookup and identification. It will be appreciated that other implementations of the system 300 are also possible. An objective of the system 300 is to minimize and prevent false positives such as shown in illustration 100 of Figure 1.

[0070] The target face 302 may be a face depicted in a target face image that is presented and input to the system 300 for authentication purposes. The target face image may be obtained, for example, in real time or recorded from a video camera. The target face image may be a digital image, a still video frame, a physical photograph, or other similar formats. The plurality of face images may be images each depicting a face, that are stored in a database or image repository. For example, the plurality of face images may be face images of customers of a commercial entity such as a payment card issuer, and a face of a customer (e.g. presented to the system 300 as target face 302) may be identified with reference to these face images before the customer is allowed to make payment or enjoy a discount for the payment using a payment card issued by the commercial entity. The plurality of face images may also be face images of employees of a company, and a face of an employee (e.g. presented to the system 300 as target face 302) may be identified with reference to these face images before the employee is allowed to enter the company building. It will be appreciated that target face image and the plurality of face images are not limited to the examples described above.

[0071] The plurality of models 312 may each comprise a different face matching algorithm which is used to identify matching face images from the plurality of face images, wherein each face depicted in the matching face images is considered by the associated face matching algorithm to be matching with the target face 302. This process comprises comparing, by each of the different face matching algorithm, the target face with each face depicted in the plurality of face images. In this manner, a plurality of matching face images can be identified by the plurality of models 312.

[0072] In the system 300, the plurality of models 312 comprises 4 different models running independently and in parallel with one another, wherein each model utilizes a different face matching algorithm. For example, the plurality of models 312 may comprise 4 parallel implementations of face-matching including a face matching algorithm from an Asia-based

vendor, a face matching algorithm from a US-based vendor, an internal implementation comprising a face matching algorithm that is developed in-house by an entity or user utilizing the system 300, and another internal implementation based on, for example, a Siamese Neural Network. A Siamese Neural Network is a special type of neural network where an exact same network is duplicated, and then the original and duplicate networks (also termed twin networks) are each given a different input. If both inputs are from a same person's face (e.g. each input depicting a different view or expression of a same person's face), the expectation is that output from each of these twin networks will be very similar to each other. If both inputs depict faces of different persons, the output will be fairly different. These networks are trained in pairs on input images from a same person, as well as input images from different persons, enabling post training networks to gain not only an understanding on how face images of a same person can look slightly different, but also that face images of different persons can have certain facial differences resulting in very different vectors. It will be appreciated that other variations of such implementations are possible and the plurality of models 312 may also implement a different number of models. Multiple parallel algorithms can advantageously improve gap areas in which certain face recognition algorithms may fail, and may rectify biases in a training dataset when training the system 300, through leveraging an ensemble of different face recognition algorithms such as shown in the plurality of models 312.

[0073] The system 300 may further comprise a plurality of lookup models running independently of and in parallel with one another. Each of the plurality of lookup models may be configured to compare the target face 302 with each of the plurality of face images to determine which of the plurality of face images depict a face that is similar to the target face. Similar to the plurality of models 312, the plurality of lookup models may comprise different implementations for looking up the plurality of face images to determine depicted faces that are similar to the target face 302. For example, the plurality of lookup models may comprise a lookup model 304 from a vendor. The lookup model 304 may also be an in-house model utilizing a proprietary algorithm for face lookup, or other commercially available algorithms for face lookup. The plurality of lookup models may also comprise a plurality of in-house lookup models 306, each utilizing a different face vectorization algorithm for vectorizing the target face. The lookup models 306 may also be from commercial vendors utilizing other commercially available or proprietary algorithms for face vectorization and/or lookup. For example, the process of determining which of the plurality of face images depict a face that is similar to the target face may further comprise vectorizing the

target face and comparing the vectorized target face with each face of the plurality of face images by each of the plurality of in-house lookup models 306, wherein each face depicted in the plurality of face images is vectorized. In an implementation, each of the plurality of in-house lookup models 306 may be configured to vectorise the target face 302 as well as each face of the plurality of face images for the comparison. In an implementation, instead of each in-house lookup model performing both vectorization of the faces and comparison of the vectorized faces, the vectorized faces from each in-house lookup model may be consolidated and compared via a vector lookup module 308. It will be appreciated that other variations of such implementations are possible and the plurality of lookup models 304 and 306 may also implement a different number of models. Advantageously, multiple parallel algorithms utilized for the face lookup process can improve gap areas in which certain face lookup algorithms may fail, and may rectify biases in a training dataset when training the system 300, through leveraging an ensemble of different face lookup algorithms such as shown in the plurality of lookup models.

[0074] Mismatches, particularly false positives that were not of a super high confidence level (e.g. a confidence level that a reference image is matching with the target image), may be reduced dramatically by following up the lookup process of the plurality of lookup models with a one-to-one (1-1) matching (e.g. an exact face match between two different face pictures that is performed by the plurality of models 312). It may also be noted that usage of extremely high confidence lookups may increase false negatives, and for certain use cases, high false negatives or low recall are not desirable. For “very high confidence” lookups, research has shown that the lookup models 304 and 306 produced 3% false positives, and 100% of these false positives were eliminated by a follow up 1-1 match. For “high confidence” match (which is a tier lower than “very high confidence” match), both lookup models 304 and 306 suffered from a large number of false positives. A follow up 1-1 match (for example, utilizing just the face matching algorithm from the above-mentioned US-based vendor) was shown to reduce false positive count by 50%. Additional 1-1 match algorithms are likely to decrease the false positive count even further.

[0075] Thus, after the lookup process of the plurality of lookup models (e.g. after determining which of the plurality of face images depict a face that is similar to the target face 302), the results of the determination (e.g. the determined face images) may be consolidated by a collection module 310 and sent to the plurality of models 312. Identifying the

plurality of matching face images by the plurality of models 312 may then be based on comparing the target face 302 with each face depicted in the determined face images.

[0076] The identification results (e.g. the plurality of matching face images) from the plurality of models 312 may be collected by an ensemble model 314. The ensemble model may comprise a neural network-based algorithm that enables prediction of a likelihood that a matching face image is a false positive, and may be configured to determine which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, by comparing each of the plurality of matching face images with one another. The ensemble model 314 may determine that a face depicted in a matching face image is an exact match with the target face 302 if the matching face image is one that is identified by all of the plurality of models. One way to achieve this may be a “logical AND” implementation of all interim output nodes of the neural network-based algorithm ensemble model 312, whereby the ensemble model 314 will only determine that a face match for the target face 302 is found if all of the plurality of models 312 have determined that a given face from the plurality of face images is a strong match to the target face. This implementation may advantageously be useful for eliminating false positives for face-based authentication. An output face image may then be generated by the ensemble model, wherein the output face image is one of the plurality of matching face images that is determined by the ensemble model as depicting a face that is an exact match with the target face. Thus, from all of the different implementations in the plurality of models 312, the ensemble model 314 generates a final output 316 of whether a face match for the target face 302 is found from the plurality of face images.

[0077] In an implementation, a multi arm bandit (e.g. reinforcement learning) configuration may be utilized for weight optimization of the plurality of models 312. For example, weights may be assigned to each of the plurality of models, wherein the determination by the ensemble model 314 (e.g. of which of the plurality of matching face images from the plurality of models 312 depict a face that is an exact match with the target face) is based on the assigned weights. Training data for the model may be provided by auxiliary human evaluation. For example, training data may be generated based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image. This comparison may be performed with human evaluation in a step 318 as a quality safeguard. This is advantageously useful when high recall values are desired, for example in applications involving face search for a criminal offender. The assignment of

weights may be based on the generated training data. Further, the training data may be utilized for retraining the ensemble model 314 in a step 320 via reinforcement learning, and the assignment of weights may be optimized based on the retraining.

5 **[0078]** In an implementation, a retraining module may also be utilized for retraining the system 300. The retraining module may be an offline component that may be configured to generate a sampled set of outcomes (e.g. comprising one or more face images from the plurality of face images that are determined by the retraining module to be an exact match with an input face) for human evaluation. Once each of the sampled set of outcomes
10 are labeled by humans (e.g. whether each outcome is a true match with the input image or a false positive), the data and labelling results are then utilized to retrain and optimize the assigned weights. The data may also be used to benchmark all of the core algorithms utilized for lookup/search and matching (e.g. the algorithms of the each of the plurality of lookup models 304, 306, and the plurality of models 312).

15

[0079] Fig. 5 illustrates an example flow diagram of how a target face may be identified from a plurality of face images using a one-to-one (1-1) match implementation according to various embodiments. In step 502, a plurality of matching face images are identified from a plurality of face images based on a target face using a plurality of models running
20 independently of and in parallel with one another, each of the plurality of models being configured to match the target face with each face depicted in the plurality of face images, each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face. In step 504, it is determined, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being
25 configured to compare each of the plurality of matching face images with one another for the determination. The plurality of models implementing the one-to-one (1-1) match may be the plurality of models 312 of system 300, and the ensemble model may refer to the ensemble model 314 of system 300.

30

[0080] Fig. 6 illustrates an example flow diagram of how a target face may be identified from a plurality of face images using a combined one-to-many (1-n) match and one-to-one (1-1) match implementation according to various embodiments. In a step 602, it is determined, by a plurality of lookup models running independently of and in parallel with one
35 another, which of the plurality of face images depict a face that is similar to the target face,

each of the plurality of lookup models being configured to compare the target face with each of the plurality of face images. The comparison may be based on a one-to-many (1-n) match implementation by the plurality of lookup models (e.g. lookup models 304 and 306). In a step 604, the plurality of matching face images are identified based on comparing the target face with each face depicted in the determined face images. The identification may be based on a one-to-one (1-1) match implementation using, for example, the plurality of models 312.

[0081] Fig. 7 illustrates an example flow diagram of how retraining of an ensemble model such as ensemble model 314 may be implemented according to various embodiments. In a step 702, an output face image is generated by the ensemble model, the output face image being one of the plurality of matching face images that is determined by the ensemble model as depicting a face that is an exact match with the target face. In a step 704, training data is generated based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image. In a step 706, weights are assigned to each of the plurality of models based on the training data, wherein the determination is based on the assigned weights. In a step 708, the ensemble model is retrained based on the training data by reinforcement learning. In a step 710, the assignment of weights is optimized based on the retraining.

[0082] Figs. 8A and 8B form a schematic block diagram of a general purpose computer system upon which the transaction processing server of Fig. 1 can be practiced.

[0083] As seen in Fig. 8A, the computer system 1300 includes a computer module 1301. An external Modulator-Demodulator (Modem) transceiver device 1316 may be used by the computer module 1301 for communicating to and from a communications network 1320 via a connection 1321. The communications network 1320 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 1321 is a telephone line, the modem 1316 may be a traditional "dial-up" modem. Alternatively, where the connection 1321 is a high capacity (e.g., cable) connection, the modem 1316 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1320.

[0084] The computer module 1301 typically includes at least one processor unit 1305, and a memory unit 1306. For example, the memory unit 1306 may have semiconductor

random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1301 also includes an interface 1308 for the external modem 1316. In some implementations, the modem 1316 may be incorporated within the computer module 1301, for example within the interface 1308. The computer module 1301 also has a local network interface 1311, which permits coupling of the computer system 1300 via a connection 1323 to a local-area communications network 1322, known as a Local Area Network (LAN). As illustrated in Fig. 8A, the local communications network 1322 may also couple to the wide network 1320 via a connection 1324, which would typically include a so-called "firewall" device or device of similar functionality. The local network interface 1311 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1311.

[0085] The I/O interfaces 1308 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1309 are provided and typically include a hard disk drive (HDD) 1310. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 1312 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks, USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the system 1300.

[0086] The components 1305 to 1312 of the computer module 1301 typically communicate via an interconnected bus 1304 and in a manner that results in a conventional mode of operation of the computer system 1300 known to those in the relevant art. For example, the processor 1305 is coupled to the system bus 1304 using a connection 1318. Likewise, the memory 1306 and optical disk drive 1312 are coupled to the system bus 1304 by connections 1319. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple or like computer systems.

[0087] The steps of the methods 500, 600 and 700 in Figs. 5, 6 and 7 facilitated by the transaction processing server 108 may be implemented using the computer system 1300. For example, the steps of the method 500 may be implemented as one or more software

application programs 1333 executable within the computer system 1300. In particular, the steps of the method 500 as facilitated by the transaction processing server 108 are effected by instructions 1331 (see Fig. 6B) in the software 1333 that are carried out within the computer system 1300. The software instructions 1331 may be formed as one or more
 5 code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules facilitates the steps of the method 500 and a second part and the corresponding code modules manage a user interface between the first part and the user.

10 **[0088]** The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the computer system 1300 from the computer readable medium, and then executed by the computer system 1300. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the
 15 computer program product in the computer system 1300 preferably effects an advantageous apparatus for a transaction processing server 108.

[0089] The software 1333 is typically stored in the HDD 1310 or the memory 1306. The software is loaded into the computer system 1300 from a computer readable medium, and
 20 executed by the computer system 1300. Thus, for example, the software 1333 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1325 that is read by the optical disk drive 1312. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer system 1300 preferably effects an apparatus for a trans-
 25 action processing server 108.

[0090] In some instances, the application programs 1333 may be supplied to the user encoded on one or more CD-ROMs 1325 and read via the corresponding drive 1312, or alternatively may be read by the user from the networks 1320 or 1322. Still further, the
 30 software can also be loaded into the computer system 1300 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the computer system 1300 for execution and/or processing. Examples of such storage media include floppy disks, magnetic tape, optical disk, a hard disk drive, a ROM or integrated circuit, USB memory, a
 35 magneto-optical disk, or a computer readable card such as a PCMCIA card and the like,

whether or not such devices are internal or external of the computer module 1301. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1301 include radio or infra-red transmission channels as well as a
 5 network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[0091] The second part of the application programs 1333 and the corresponding code modules mentioned above may be executed to implement one or more graphical user
 10 interfaces (GUIs) to be rendered or otherwise represented upon a display. Through manipulation of typically a keyboard and a mouse, a user of the computer system 1300 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an au-
 15 dio interface utilizing speech prompts output via loudspeakers and user voice commands input via a microphone.

[0092] Fig. 8B is a detailed schematic block diagram of the processor 1305 and a “memory” 1334. The memory 1334 represents a logical aggregation of all the memory
 20 modules (including the HDD 1309 and semiconductor memory 1306) that can be accessed by the computer module 1301 in Fig. 8A.

[0093] When the computer module 1301 is initially powered up, a power-on self-test (POST) program 1350 executes. The POST program 1350 is typically stored in a
 25 ROM 1349 of the semiconductor memory 1306 of Fig. 8A. A hardware device such as the ROM 1349 storing software is sometimes referred to as firmware. The POST program 1350 examines hardware within the computer module 1301 to ensure proper functioning and typically checks the processor 1305, the memory 1334 (1309, 1306), and a basic input-output systems software (BIOS) module 1351, also typically stored in the
 30 ROM 1349, for correct operation. Once the POST program 1350 has run successfully, the BIOS 1351 activates the hard disk drive 1310 of Fig. 8A. Activation of the hard disk drive 1310 causes a bootstrap loader program 1352 that is resident on the hard disk drive 1310 to execute via the processor 1305. This loads an operating system 1353 into the RAM memory 1306, upon which the operating system 1353 commences operation.

The operating system 1353 is a system level application, executable by the processor 1305, to fulfil various high level functions, including processor management, memory management, device management, storage management, software application interface, and generic user interface.

5

[0094] The operating system 1353 manages the memory 1334 (1309, 1306) to ensure that each process or application running on the computer module 1301 has sufficient memory in which to execute without colliding with memory allocated to another process. Furthermore, the different types of memory available in the system 1300 of Fig. 8A must be used properly so that each process can run effectively. Accordingly, the aggregated memory 1334 is not intended to illustrate how particular segments of memory are allocated (unless otherwise stated), but rather to provide a general view of the memory accessible by the computer system 1300 and how such is used.

10

15

[0095] As shown in Fig. 8B, the processor 1305 includes a number of functional modules including a control unit 1339, an arithmetic logic unit (ALU) 1340, and a local or internal memory 1348, sometimes called a cache memory. The cache memory 1348 typically includes a number of storage registers 1344 - 1346 in a register section. One or more internal busses 1341 functionally interconnect these functional modules. The processor 1305 typically also has one or more interfaces 1342 for communicating with external devices via the system bus 1304, using a connection 1318. The memory 1334 is coupled to the bus 1304 using a connection 1319.

20

25

[0096] The application program 1333 includes a sequence of instructions 1331 that may include conditional branch and loop instructions. The program 1333 may also include data 1332 which is used in execution of the program 1333. The instructions 1331 and the data 1332 are stored in memory locations 1328, 1329, 1330 and 1335, 1336, 1337, respectively. Depending upon the relative size of the instructions 1331 and the memory locations 1328-1330, a particular instruction may be stored in a single memory location as depicted by the instruction shown in the memory location 1330. Alternately, an instruction may be segmented into a number of parts each of which is stored in a separate memory location, as depicted by the instruction segments shown in the memory locations 1328 and 1329.

30

[0097] In general, the processor 1305 is given a set of instructions which are executed therein. The processor 1305 waits for a subsequent input, to which the processor 1305 reacts to by executing another set of instructions. Each input may be provided from one or more of a number of sources, including data generated by one or more of the input devices 1302, 1303, data received from an external source across one of the networks 1320, 1302, data retrieved from one of the storage devices 1306, 1309 or data retrieved from a storage medium 1325 inserted into the corresponding reader 1312, all depicted in Fig. 8A. The execution of a set of the instructions may in some cases result in output of data. Execution may also involve storing data or variables to the memory 1334.

[0098] The disclosed transaction processing server 108 arrangements use input variables 1354, which are stored in the memory 1334 in corresponding memory locations 1355, 1356, 1357. The transaction processing server 108 arrangements produce output variables 1361, which are stored in the memory 1334 in corresponding memory locations 1362, 1363, 1364. Intermediate variables 1358 may be stored in memory locations 1359, 1360, 1366 and 1367.

[0099] Referring to the processor 1305 of Fig. 8B, the registers 1344, 1345, 1346, the arithmetic logic unit (ALU) 1340, and the control unit 1339 work together to perform sequences of micro-operations needed to perform “fetch, decode, and execute” cycles for every instruction in the instruction set making up the program 1333. Each fetch, decode, and execute cycle comprises:

a fetch operation, which fetches or reads an instruction 1331 from a memory location 1328, 1329, 1330;

a decode operation in which the control unit 1339 determines which instruction has been fetched; and

an execute operation in which the control unit 1339 and/or the ALU 1340 execute the instruction.

[0100] Thereafter, a further fetch, decode, and execute cycle for the next instruction may be executed. Similarly, a store cycle may be performed by which the control unit 1339 stores or writes a value to a memory location 1332.

[0101] Each step or sub-process in the processes as performed by the transaction processing server 108, is associated with one or more segments of the program 1333 and is performed by the register section 1344, 1345, 1347, the ALU 1340, and the control unit 1339 in the processor 1305 working together to perform the fetch, decode, and execute cycles for every instruction in the instruction set for the noted segments of the program 1333.

[0102] It is to be understood that the structural context of the computer system 1300 (i.e., the transaction processing server 108) is presented merely by way of example. Therefore, in some arrangements, one or more features of the server 1300 may be omitted. Also, in some arrangements, one or more features of the server 1300 may be combined together. Additionally, in some arrangements, one or more features of the server 1300 may be split into one or more component parts.

[0103] Fig. 9 shows an alternative implementation of the transaction processing server 108 (i.e., the computer system 1300). In the alternative implementation, the transaction processing 108 may be generally described as a physical device comprising at least one processor 802 and at least one memory 804 including computer program codes. The at least one memory 804 and the computer program codes are configured to, with the at least one processor 802, cause the transaction processing server 108 to facilitate the operations described in the methods 500, 600 and 700. The transaction processing server 108 may also include a transaction processing module 806. The memory 804 stores computer program code that the processor 802 compiles to have each of the modules 806 and 808 performs their respective functions.

[0104] With reference to Fig. 1, the transaction processing module 806 performs the function of communicating with the requestor device 102 and the provider device 104; and the acquirer server 106 and the issuer server 110 to respectively receive and transmit a transaction or travel request message. Further, the transaction processing module 806 may provide data and information associated with the target image and plurality of images that are used for the face recognition process of face recognition server 140. The transaction or travel request message may then be authenticated based on an outcome of the face recognition process e.g. when the target face depicted in the target image is identified as one of the faces depicted in the plurality of images, and data and information associated

with the identified face corresponds with data and information associated with the user of the requestor device 102 or the provider device 104.

[0105] Fig. 8C depict a general-purpose computer system 1400, upon which the face recognition server 140 described can be practiced. The computer system 1400 includes a computer module 1401. An external Modulator-Demodulator (Modem) transceiver device 1416 may be used by the computer module 1401 for communicating to and from a communications network 1420 via a connection 1421. The communications network 1420 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 1421 is a telephone line, the modem 1416 may be a traditional “dial-up” modem. Alternatively, where the connection 1421 is a high capacity (e.g., cable) connection, the modem 1416 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1420.

[0106] The computer module 1401 typically includes at least one processor unit 1405, and a memory unit 1406. For example, the memory unit 1406 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1401 also includes an interface 1408 for the external modem 1416. In some implementations, the modem 1416 may be incorporated within the computer module 1401, for example within the interface 1408. The computer module 1401 also has a local network interface 1411, which permits coupling of the computer system 1400 via a connection 1423 to a local-area communications network 1422, known as a Local Area Network (LAN). As illustrated in Fig. 8C, the local communications network 1422 may also couple to the wide network 1420 via a connection 1424, which would typically include a so-called “firewall” device or device of similar functionality. The local network interface 1411 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1411.

[0107] The I/O interfaces 1408 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1409 are provided and typically include a hard disk drive (HDD) 1410. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used.

An optical disk drive 1412 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks, USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the system 1400.

5

[0108] The components 1405 to 1412 of the computer module 1401 typically communicate via an interconnected bus 1404 and in a manner that results in a conventional mode of operation of the computer system 1400 known to those in the relevant art. For example, the processor 1405 is coupled to the system bus 1404 using a connection 1418. Likewise, the memory 1406 and optical disk drive 1412 are coupled to the system bus 1404 by connections 1419. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple or like computer systems.

[0109] The method 500, where performed by the face recognition server 140 may be implemented using the computer system 1400. The processes may be implemented as one or more software application programs 1433 executable within the computer system 1400. In particular, the sub-processes 400, 500, and 600 are effected by instructions (see corresponding component 1331 in Fig. 8B) in the software 1433 that are carried out within the computer system 1400. The software instructions may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the methods and a second part and the corresponding code modules manage a user interface between the first part and the user.

25

[0110] The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the computer system 1400 from the computer readable medium, and then executed by the computer system 1400. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the computer program product in the computer system 1400 preferably effects an advantageous apparatus for a face recognition server 140.

30

[0111] The software 1433 is typically stored in the HDD 1410 or the memory 1406. The software is loaded into the computer system 1400 from a computer readable medium, and

35

executed by the computer system 1400. Thus, for example, the software 1433 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1425 that is read by the optical disk drive 1412. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer system 1400 preferably effects an apparatus for a face recognition server 140.

[0112] In some instances, the application programs 1433 may be supplied to the user encoded on one or more CD-ROMs 1425 and read via the corresponding drive 1412, or alternatively may be read by the user from the networks 1420 or 1422. Still further, the software can also be loaded into the computer system 1400 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the computer system 1400 for execution and/or processing. Examples of such storage media include floppy disks, magnetic tape, optical disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 1401. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1401 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[0113] The second part of the application programs 1433 and the corresponding code modules mentioned above may be executed to implement one or more graphical user interfaces (GUIs) to be rendered or otherwise represented upon a display. Through manipulation of typically a keyboard and a mouse, a user of the computer system 1400 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via loudspeakers and user voice commands input via a microphone.

[0114] It is to be understood that the structural context of the computer system 1400 (i.e., the face recognition server 140) is presented merely by way of example. Therefore, in

some arrangements, one or more features of the computer system 1400 may be omitted. Also, in some arrangements, one or more features of the computer system 1400 may be combined together. Additionally, in some arrangements, one or more features of the computer system 1400 may be split into one or more component parts.

5

[0115] Fig. 10 shows an alternative implementation of the face recognition server 140 (i.e., the computer system 1400). In the alternative implementation, face recognition server 140 may be generally described as a physical device comprising at least one processor 902 and at least one memory 904 including computer program codes. The at least one memory 904 and the computer program codes are configured to, with the at least one processor 902, cause the face recognition server 140 to perform the operations described in the methods 500, 600 and 700. The face recognition server 140 may also include an ensemble module 906, a data module 908, a collection module 910, a retraining module 912, a lookup module 914 (e.g. comprising the plurality of lookup models 262) and a matching module 916 (e.g. comprising the plurality of models 264). The memory 904 stores computer program code that the processor 902 compiles to have each of the modules 906 to 916 performs their respective functions.

[0116] With reference to Figs. 1 to 3 and 5, the matching module 914 performs the function of identifying a plurality of matching face images from a plurality of face images based on a target face using a plurality of models running independently of and in parallel with one another, each of the plurality of models being configured to match the target face with each face depicted in the plurality of face images, each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face.

[0117] With reference to Figs. 1 to 3 and 5, the ensemble module 906 performs the function of determining, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being configured to compare each of the plurality of matching face images with one another for the determination.

[0118] With reference to Figs. 1 to 3 and 6, the lookup module 914 performs the function of determining which of the plurality of face images depict a face that is similar to the target face using a plurality of lookup models running independently of and in parallel with one

another, each of the plurality of lookup models being configured to compare the target face with each of the plurality of face images, and identifying the plurality of matching face images based on comparing the target face with each face depicted in the determined face images.

5

[0119] With reference to Figs. 1 to 3 and 6, the collection module 910 performs the function of consolidating the plurality of images that are determined by the plurality of lookup models to depict a face that is similar to the target face, and providing the consolidated images as input into each of the plurality of models of the matching module 914.

10

[0120] With reference to Figs. 1 to 3 and 7, the ensemble module 906 performs the function of generating, by the ensemble model, an output face image, the output face image being one of the plurality of matching face images that is determined by the ensemble model as depicting a face that is an exact match with the target face.

15

[0121] With reference to Figs. 1 to 3 and 7, the retraining module 912 performs the function of generating training data based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image, and retrain the ensemble model based on the training data by reinforcement learning. The retraining module 912 may also perform the functions of assigning weights to each of the plurality of models based on the training data, wherein the determination is based on the assigned weights, and optimizing the assignment of weights based on the retraining.

20

[0122] With reference to Figs. 1 to 3 and 5 to 7, the data module 908 performs the functions of receiving data and information from the requestor device 102, provider device 104, transaction processing server 108, a cloud and other sources of information to facilitate the methods 500, 600 and 700. For example, the data module 908 may be configured to receive a target image depicting a target face from the requestor device 102, the provider device 104, transaction processing server 108 or other sources of information. The data module 908 may also be configured to receive a plurality of face images each depicting a face from the transaction processing server 108, the reference image database 150 or other sources of information. The data module 260 may be further configured to send an image such as an output image obtained after a face recognition process (e.g. from the ensemble module 906) to the transaction processing server 108, a database or other sources of information.

25

30

35

[0123] Fig. 8D depicts a general-purpose computer system 1500, upon which a combined transaction processing server 108 and face recognition server 140 described can be practiced. The computer system 1500 includes a computer module 1501. An external Modulator-Demodulator (Modem) transceiver device 1516 may be used by the computer module 1501 for communicating to and from a communications network 1520 via a connection 1521. The communications network 1520 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 1521 is a telephone line, the modem 1516 may be a traditional “dial-up” modem. Alternatively, where the connection 1521 is a high capacity (e.g., cable) connection, the modem 1516 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1520.

[0124] The computer module 1501 typically includes at least one processor unit 1505, and a memory unit 1506. For example, the memory unit 1506 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1501 also includes an interface 1508 for the external modem 1516. In some implementations, the modem 1516 may be incorporated within the computer module 1501, for example within the interface 1508. The computer module 1501 also has a local network interface 1511, which permits coupling of the computer system 1500 via a connection 1523 to a local-area communications network 1522, known as a Local Area Network (LAN). As illustrated in Fig. 8D, the local communications network 1522 may also couple to the wide network 1520 via a connection 1524, which would typically include a so-called “firewall” device or device of similar functionality. The local network interface 1511 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1511.

[0125] The I/O interfaces 1508 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1509 are provided and typically include a hard disk drive (HDD) 1510. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 1512 is typically provided to act as a non-volatile source of data. Portable memory devices, such as optical disks, USB-RAM, portable, external hard drives,

and floppy disks, for example, may be used as appropriate sources of data to the system 1500.

[0126] The components 1505 to 1512 of the computer module 1501 typically communicate via an interconnected bus 1504 and in a manner that results in a conventional mode of operation of the computer system 1500 known to those in the relevant art. For example, the processor 1505 is coupled to the system bus 1504 using a connection 1518. Likewise, the memory 1506 and optical disk drive 1512 are coupled to the system bus 1504 by connections 1519. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple or like computer systems.

[0127] The steps of the methods 500, 600 and 700 performed by the face recognition server 140 and facilitated by the transaction processing server 108 may be implemented using the computer system 1500. For example, the steps of the method 500 as performed by the face recognition server 140 may be implemented as one or more software application programs 1533 executable within the computer system 1500. In particular, the steps of the methods 500, 600 and 700 are effected by instructions (see corresponding component 1331 in Fig. 8B) in the software 1533 that are carried out within the computer system 1500. The software instructions may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the steps of the methods 500, 600 and 700 and a second part and the corresponding code modules manage a user interface between the first part and the user.

[0128] The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the computer system 1500 from the computer readable medium, and then executed by the computer system 1500. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the computer program product in the computer system 1500 preferably effects an advantageous apparatus for a combined transaction processing server 108 and face recognition server 140.

[0129] The software 1533 is typically stored in the HDD 1510 or the memory 1506. The software is loaded into the computer system 1500 from a computer readable medium, and executed by the computer system 1500. Thus, for example, the software 1533 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1525 that is read by the optical disk drive 1512. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the computer system 1500 preferably effects an apparatus for a combined transaction processing server 108 and face recognition server 140.

[0130] In some instances, the application programs 1533 may be supplied to the user encoded on one or more CD-ROMs 1525 and read via the corresponding drive 1512, or alternatively may be read by the user from the networks 1520 or 1522. Still further, the software can also be loaded into the computer system 1500 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the computer system 1500 for execution and/or processing. Examples of such storage media include floppy disks, magnetic tape, optical disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 1501. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1501 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[0131] The second part of the application programs 1533 and the corresponding code modules mentioned above may be executed to implement one or more graphical user interfaces (GUIs) to be rendered or otherwise represented upon a display. Through manipulation of typically a keyboard and a mouse, a user of the computer system 1500 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via loudspeakers and user voice commands input via a microphone.

[0132] It is to be understood that the structural context of the computer system 1500 (i.e., combined transaction processing server 108 and face recognition server 140) is presented merely by way of example. Therefore, in some arrangements, one or more features of the server 1500 may be omitted. Also, in some arrangements, one or more features of the server 1500 may be combined together. Additionally, in some arrangements, one or more features of the server 1500 may be split into one or more component parts.

[0133] Fig. 11 shows an alternative implementation of combined transaction processing server 108 and face recognition server 140 (i.e., the computer system 1500). In the alternative implementation, the combined transaction processing server 108 and face recognition server 140 may be generally described as a physical device comprising at least one processor 1002 and at least one memory 904 including computer program codes. The at least one memory 1004 and the computer program codes are configured to, with the at least one processor 1002, cause the combined transaction processing server 108 and face recognition server 140 to perform the operations described in the steps of the methods 500, 600 and 700. The combined transaction processing server 108 and face recognition server 140 may also include a transaction request processing module 806, an ensemble module 906, a data module 908, a collection module 910, a retraining module 912, a lookup module 914 (e.g. comprising the plurality of lookup models 262) and a matching module 916 (e.g. comprising the plurality of models 264). The memory 1004 stores computer program code that the processor 1002 compiles to have each of the modules 806 to 912 performs their respective functions.

[0134] With reference to Fig. 1, the transaction processing module 806 performs the function of communicating with the requestor device 102 and the provider device 104; and the acquirer server 106 and the issuer server 110 to respectively receive and transmit a transaction or travel request message. Further, the transaction processing module 806 may provide data and information associated with the target image and plurality of images that are used for the face recognition process of face recognition server 140. The transaction or travel request message may then be authenticated based on an outcome of the face recognition process e.g. when the target face depicted in the target image is identified as one of the faces depicted in the plurality of images, and data and information associated with the identified face corresponds with data and information associated with the user of the requestor device 102 or the provider device 104.

[0135] With reference to Figs. 1 to 3 and 5, the matching module 914 performs the function of identifying a plurality of matching face images from a plurality of face images based on a target face using a plurality of models running independently of and in parallel with one another, each of the plurality of models being configured to match the target face with each face depicted in the plurality of face images, each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face.

[0136] With reference to Figs. 1 to 3 and 5, the ensemble module 906 performs the function of determining, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being configured to compare each of the plurality of matching face images with one another for the determination.

[0137] With reference to Figs. 1 to 3 and 6, the lookup module 914 performs the function of determining which of the plurality of face images depict a face that is similar to the target face using a plurality of lookup models running independently of and in parallel with one another, each of the plurality of lookup models being configured to compare the target face with each of the plurality of face images, and identifying the plurality of matching face images based on comparing the target face with each face depicted in the determined face images.

[0138] With reference to Figs. 1 to 3 and 6, the collection module 910 performs the function of consolidating the plurality of images that are determined by the plurality of lookup models to depict a face that is similar to the target face, and providing the consolidated images as input into each of the plurality of models of the matching module 914.

[0139] With reference to Figs. 1 to 3 and 7, the ensemble module 906 performs the function of generating, by the ensemble model, an output face image, the output face image being one of the plurality of matching face images that is determined by the ensemble model as depicting a face that is an exact match with the target face.

[0140] With reference to Figs. 1 to 3 and 7, the retraining module 912 performs the function of generating training data based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image, and retrain

the ensemble model based on the training data by reinforcement learning. The retraining module 912 may also perform the functions of assigning weights to each of the plurality of models based on the training data, wherein the determination is based on the assigned weights, and optimizing the assignment of weights based on the retraining.

5

[0141] With reference to Figs. 1 to 3 and 5 to 7, the data module 908 performs the functions of receiving data and information from the requestor device 102, provider device 104, a cloud and other sources of information to facilitate the methods 500, 600 and 700. For example, the data module 908 may be configured to receive a target image depicting a target face from the requestor device 102, the provider device 104 or other sources of information. The data module 908 may also be configured to receive a plurality of face images each depicting a face from the reference image database 150 or other sources of information. The data module 260 may be further configured to send an image such as an output image obtained after a face recognition process (e.g. from the ensemble module 906) to a database or other sources of information.

10
15

[0142] It will be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present disclosure as shown in the specific embodiments without departing from the scope of the specification as broadly described. The present embodiments are, therefore, to be considered in all respects to be illustrative and not restrictive.

20

CLAIMS

What is claimed is:

1. A method for identifying a target face depicted in a target face image from a plurality of face images, each of the plurality of face images depicting a face, comprising:
5 identifying a plurality of matching face images from the plurality of face images based on the target face using a plurality of models running independently of and in parallel with one another, each of the plurality of models being configured to compare the target face with each face depicted in the plurality of face images,
10 each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face; and
determining, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being configured to compare each of the plurality
15 of matching face images with one another for the determination, wherein the ensemble model determines that a face depicted in a matching face image is an exact match with the target face if the matching face image is one that is identified by all of the plurality of models.
- 20 2. The method of claim 1, further comprising determining which of the plurality of face images depict a face that is similar to the target face using a plurality of lookup models running independently of and in parallel with one another, each of the plurality of lookup models being configured to compare the target face with each of the plurality of face images, wherein identifying the plurality of matching
25 face images is based on comparing the target face with each face depicted in the determined face images.
3. The method of claim 2, wherein each of the plurality of lookup models comprise a different face vectorization algorithm for vectorizing the target face, wherein determining which of the plurality of face images depict a face that is similar to the
30 target face further comprises vectorizing the target face and comparing the vectorized target face with each face of the plurality of face images by each of the plurality of lookup models, wherein each face depicted in the plurality of face images is vectorised.

4. The method of claim 1, wherein each of the plurality of models comprise a different face matching algorithm, wherein identifying matching face images from the plurality of face images further comprises comparing, by each of the different face matching algorithm, the target face with each face depicted in the plurality of face images.
5. The method of claim 1, further comprising:
assigning weights to each of the plurality of models, wherein the determination is based on the assigned weights.
6. The method of claim 1, further comprising:
generating, by the ensemble model, an output face image, the output face image being one of the plurality of matching face images that is determined by the ensemble model as depicting a face that is an exact match with the target face.
7. The method of claim 6, further comprising:
generating training data based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image; and
assigning weights to each of the plurality of models based on the training data, wherein the determination is based on the assigned weights.
8. The method of claim 7, further comprising:
retraining the ensemble model based on the training data by reinforcement learning; and
optimizing the assignment of weights based on the retraining.
9. A system for identifying a target face depicted in a target face image from a plurality of face images, each of the plurality of face images depicting a face, the system comprising:
at least one processor; and
at least one memory including computer program code;
the at least one memory and the computer program code configured to, with the at least one processor, cause the system at least to:

identify a plurality of matching face images from the plurality of face images based on the target face using a plurality of models running independently of and in parallel with one another, each of the plurality of models being configured to compare the target face with each face depicted in the plurality of face images, each of the plurality of matching face images being identified by the plurality of models as depicting a face that matches with the target face; and

determine, by an ensemble model, which of the plurality of matching face images from the plurality of models depict a face that is an exact match with the target face, the ensemble model being configured to compare each of the plurality of matching face images with one another for the determination, wherein the ensemble model determines that a face depicted in a matching face image is an exact match with the target face if the matching face image is one that is identified by all of the plurality of models.

10. The system of claim 9, further configured to determine which of the plurality of face images depict a face that is similar to the target face using a plurality of lookup models running independently of and in parallel with one another, each of the plurality of lookup models being configured to compare the target face with each of the plurality of face images, wherein identifying the plurality of matching face images is based on comparing the target face with each face depicted in the determined face images.

11. The system to claim 10, wherein each of the plurality of lookup models comprise a different face vectorization algorithm for vectorizing the target face, wherein determining which of the plurality of face images depict a face that is similar to the target face further comprises vectorizing the target face and comparing the vectorized target face with each face depicted in the plurality of face images by each of the plurality of lookup models, wherein each face depicted in the plurality of face images is vectorised.

12. The system of claim 9, wherein each of the plurality of models comprises a different face matching algorithm, wherein identifying the plurality of matching face images from the plurality of face images further comprises comparing, by each of the different face matching algorithms, the target face with each face depicted in the plurality of face images.

13. The system of claim 9, further configured to assign weights to each of the plurality of models, wherein the determination is based on the assigned weights.

5 14. The system of claim 9, further configured to:
 generate, by the ensemble model, an output face image, the output face image being one of the plurality of matching face images that is determined by the ensemble model as depicting a face that is an exact match with the target face.

10 15. The system of claim 14, further configured to:
 generate training data based on a comparison of each face depicted in the identified matching face images with the face depicted in the output face image;
 and
 assign weights to each of the plurality of models based on the training data,
15 wherein the determination is based on the assigned weights.

 16. The system of claim 15, further configured to:
 retrain the ensemble model based on the training data by reinforcement learning; and
20 optimize the assignment of weights based on the retraining.