

METHOD AND SYSTEM FOR PROCESSING REFERENCE FACES

TECHNICAL FIELD

[001] The present specification relates broadly, but not exclusively, to a method and system for processing reference faces for face recognition.

BACKGROUND

[002] Face recognition is the process of identifying or verifying the identity of a person based on his/her face. In a face recognition based login or payment scenario, a face recognition system may first save one or more face images of a user as reference faces for verification purpose. When data of a new face is received, the system may compare the new face with the reference faces to verify if the new face and the reference faces belong to the same identity. Reference face images are therefore stored in the system in such implementations, which may potentially give rise to privacy breach when face data of the user is leaked.

[003] As an alternative to storing the original reference face images in the face recognition system, a face image may be converted to face features (e.g. in a vector form) such that the face features are stored in the system to prevent leaking of the original face images. However, there are ways to recover the original face images using face features as inputs, and the issue of privacy breach remains.

SUMMARY

[004] Described embodiments provide methods, systems, apparatuses, and devices for processing reference faces for face recognition with increased privacy protection. In some embodiments, a face recognition model can convert a user's face image to a face feature vector. A similarity score between two faces (e.g. a face to be authenticated and a reference face) can be calculated based on the two face feature vectors. In some implementations, the face recognition system may use at least two reference faces. The at least two reference faces may be converted to corresponding reference face feature vectors and encryption can be performed on

the reference face feature vectors, and the system can discard the original reference face images and the reference face feature vectors thereafter.

[005] In some embodiments, a function which calculates the similarity score of two face feature vectors is a linear function, such as a vector dot product. With two or more reference faces, two or more respective similarity scores can be calculated for an authentication input (i.e. the face to be authenticated), and an average value of the respective similarity scores can be determined as the final similarity score. In some implementations, encryption is performed on the reference face feature vectors such that the corresponding original reference faces may not be recovered from the encrypted face feature vectors. In some implementations, the encryption may involve a random number generator generating a random noise feature vector, which is used to encrypt each of the original reference face feature vectors. Subsequently, the encrypted reference face feature vectors can be stored by the face recognition system, and the original reference face feature vectors can be discarded for privacy protection.

[006] According to one embodiment, there is provided a computer-implemented method for processing reference faces. The method includes: generating a plurality of reference features corresponding to a plurality of reference faces of the user; encrypting each of the plurality of reference features to obtain a plurality of encrypted features, wherein the plurality of encrypted features are associated with the plurality of reference features such that a final similarity score between an authentication input and the plurality of reference faces of the user can be calculated based on the authentication input and the plurality of encrypted features; and discarding the plurality of reference features.

[007] According to another embodiment, there is provided a computer-implemented method for face recognition. The method includes: receiving an authentication input; determining respective similarity scores between the authentication input and a plurality of encrypted features, wherein the plurality of encrypted features are obtained by encrypting each of a plurality of reference features that correspond to a plurality of reference faces of a user; calculating, based on the respective similarity scores, a final similarity score between the authentication input and the plurality of reference faces of the user; and verifying the authentication input based on the final similarity score.

[008] According to other embodiments, one or more of these general and specific embodiments may be implemented using an apparatus including a plurality of modules, a system, a method, or a computer-readable media, or any combination of devices, systems, methods, and computer-readable media. The foregoing and other described embodiments can each, optionally, include some, none or all of the following embodiments.

BRIEF DESCRIPTION OF THE DRAWINGS

[009] 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:

[0010] Figure 1 is a flow chart illustrating an example of a computer-implemented method for processing reference faces, according to an embodiment.

[0011] Figure 2 is a flow diagram illustrating an example of an implementation of the method in Figure 1, according to an embodiment.

[0012] Figures 3 is a flow chart illustrating an example of a computer-implemented method for face recognition, according to an embodiment.

[0013] Figure 4A is a schematic diagram of an example of modules of an apparatus for processing reference faces, according to an embodiment.

[0014] Figure 4B is a schematic diagram of an example of modules of an apparatus for face recognition, according to an embodiment.

[0015] Figure 5 is a block diagram of an example of a computer system suitable for executing at least some steps of the example methods shown in Figures 1 to 3, according to an embodiment.

[0016] 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.

DETAILED DESCRIPTION

[0017] 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.

[0018] 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.

[0019] Unless specifically stated otherwise, and as apparent from the following, it will be appreciated that throughout the present specification, discussions utilizing terms such as “receiving”, “generating”, “obtaining”, “determining”, “encrypting”, “matching”, “saving”, “storing”, “discarding”, “entering”, “calculating”, “adding”, “subtracting”, “comparing”, “processing”, “training”, “updating”, “selecting”, “authenticating”, “verifying”, “providing”, “inputting”, “outputting”, 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.

[0020] The present specification also discloses apparatuses for performing the operations of the methods. Such apparatuses may be specially constructed for the required purposes, or may comprise a computer or other device selectively activated or reconfigured by a computer program stored in the computer. The algorithms and displays presented herein are not inherently related to any

particular computer or other apparatus. Various machines may be used with programs in accordance with the teachings herein. Alternatively, the construction of more specialized apparatus to perform the required method steps may be appropriate. The structure of a computer suitable for executing the various methods / processes described herein will appear from the description below.

[0021] 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 specification 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.

[0022] 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 method.

[0023] The present specification may also be implemented as hardware modules. More particularly, in the hardware sense, a module is a functional hardware unit designed for use with other components or modules. For example, a module may be implemented using discrete electronic components, or it can form a portion of an entire electronic circuit such as an Application Specific Integrated Circuit (ASIC) or Field Programmable Gate Array (FPGA). Numerous other possibilities exist. Those skilled in the art will appreciate that the system can also be implemented as a combination of hardware and software modules.

[0024] Face recognition in the act or process of identifying or verifying a person's identity can be considered as a form of identity authentication, in which legitimacy of users are verified based on their faces for security purpose. Effective identity authentication can enhance data security of systems by permitting the authenticated users to access its protected resources. Advantageously, financial risks such as money laundering and fraud can be effectively reduced or eliminated.

[0025] Embodiments described in this specification produce one or more technical effects. By encrypting the reference features of users' reference faces, potential risks of privacy breach can be prevented. Notably, original reference faces or original reference features are not saved in the face recognition system. Once encrypted features are generated based on the original reference features, the original reference faces and reference features can be discarded, and in some embodiments, corresponding data or copies of the original reference faces can also be deleted from database or memory. As the encrypted features are random, neither the original reference faces nor the original reference features can be recovered by reverse engineering and data privacy can be improved. Furthermore, a final similarity score for a face to be authenticated can be calculated based on the encrypted features, and the score can reflect accurately a similarity between the face to be authenticated and the reference faces. In some embodiments, generating the encrypted features involves encrypting the original reference features by a random noise feature, and the random noise feature can also be discarded for improving privacy once the encrypted features are generated.

[0026] Figure 1 is a flow chart 100 illustrating an example method for processing reference faces of a user. In some embodiments, at least two reference faces of a user can be used, and the example method includes the following steps:

- 110: generating a plurality of reference features corresponding to a plurality of reference faces of the user;
- 120: encrypting each of the plurality of reference features to obtain a plurality of encrypted features, wherein the plurality of encrypted features are associated with the plurality of reference features such that a final similarity score between an authentication input and the plurality of reference faces of the user can be calculated based on the authentication input and the plurality of encrypted features; and
- 130: discarding the plurality of reference features.

[0027] At step 110, a plurality of reference features corresponding to the reference faces of a user can be generated. In some embodiments, at least two reference faces of the user may be used by the face recognition system, which can be collected from the user (e.g. by a device camera) or selected from a pool of face images of the user. Upon receiving the two or more reference faces, a face recognition model can then be used to convert each of the reference faces into a corresponding feature, such as a face feature vector. Such face recognition models may include FaceNet, DeepFace, ArcFace, InsightFace, and other machine learning models. The implementations are not limited. After each reference face is converted to a corresponding reference face feature vector, the original reference faces can be discarded such that no original face images are stored by the face recognition system.

[0028] At step 120, encryption can be performed on each of the reference face feature vectors to obtain a plurality of encrypted feature vectors. Such encryption advantageously prevents reverse engineering on the reference face feature vectors to recover the original reference faces of the user. The plurality of encrypted features are associated with the plurality of reference features such that, when an input (e.g. a face to be authenticated) is received for authentication, a similarity score for the input can be calculated based on the input and the plurality of encrypted feature vectors. The similarity score can indicate a result of the face verification or face matching between the authentication input and the plurality of reference faces of the user. Notably, while each of the encrypted feature vectors differ from each of the reference face feature vectors, a final similarity score (i.e. similarity between a face to be authenticated and the reference faces of the user) may be calculated accurately using the plurality of encrypted feature vectors. This is elaborated in further details with reference to Figure 2.

[0029] In one embodiment, encrypting each of the plurality of reference face feature vectors can be performed by generating a random noise feature vector by a random number generator, and encrypting each of the plurality of reference face feature vectors based on the random noise feature vector. Encryption of the reference face feature vectors can also be performed in other ways, such as using other feature value confusion methods or using feature length confusion methods. The implementations are not limited.

[0030] At step 130, after each reference face feature vector is encrypted, the plurality of reference face feature vectors can be discarded. In implementations when encryption is done based on the random noise feature vector, the random noise feature vector can also be discarded such that the chance of obtaining the original reference faces from reverse engineering the encrypted feature vectors is minimized. Advantageously, risk of privacy breach can be alleviated since no original reference face feature vectors are stored by the face recognition system.

[0031] In some implementations, the method may proceed to step 140, at which the plurality of encrypted feature vectors can be stored by the face recognition system. One can appreciate that the encrypted feature vectors that are stored in the face recognition system can be readily used for future face recognition processes, which advantageously improve efficiency and reduce processing time. One can also appreciate that, as original face information of the user cannot be recovered from the encrypted feature vectors, storing the encrypted feature vectors does not put the user's data privacy at risk in an event of security breach.

[0032] Figure 2 is an example flow diagram 200 illustrating an implementation of the method in Figure 1. Two reference faces of a user are used in the example implementation, which are converted to two reference face feature vectors denoted as F_1 and F_2 , respectively. F_1 and F_2 are unencrypted.

[0033] Next, F_1 and F_2 are inputted into an encryptor 202. In the example implementation, the encryptor 202 can include a random number generator denoted by the function $\text{random}()$, which can generate a random noise feature vector F_d . In an example, F_1 and F_2 can be encrypted as follows: encrypt F_1 by adding the random noise feature vector F_d to F_1 , and encrypt F_2 by subtracting the random noise feature vector F_d from F_2 . As shown in the flow diagram, two encrypted feature vectors E_1 and E_2 are obtained after the encryption.

[0034] In implementations, after encryption, original feature vectors F_1 and F_2 and the random noise feature vector F_d can be discarded for privacy protection. A final similarity score between an authentication input and the two reference faces can be calculated with the encrypted feature vectors E_1 and E_2 . In the calculations below, the face feature vector corresponding to the authentication input is denoted as F_0 , and $\text{sim}()$ represents a function which calculates similarity

scores using two feature vectors as inputs. In some embodiments, $\text{sim}()$ function can be a linear function, such as a vector dot product. In relation to the two original reference features F_1 and F_2 , a final similarity score S for the authentication input can be obtained by calculating an average value of the respective similarity scores between F_0 and F_1 (S_1) and between F_0 and F_2 (S_2):

$$S_1 = \text{sim}(F_0, F_1)$$

$$S_2 = \text{sim}(F_0, F_2)$$

$$S = 0.5 \cdot (S_1 + S_2)$$

[0035] A similarity score S^e calculated based on E_1 and E_2 can be obtained by calculating an average value of the respective similarity scores between F_0 and E_1 (S^e_1) and between F_0 and E_2 (S^e_2):

$$S^e_1 = \text{sim}(F_0, E_1)$$

$$S^e_2 = \text{sim}(F_0, E_2)$$

$$S^e = 0.5 \cdot (S^e_1 + S^e_2)$$

$$= 0.5 \cdot (\text{sim}(F_0, E_1) + \text{sim}(F_0, E_2))$$

$$= 0.5 \cdot (\text{sim}(F_0, F_1 + F_d) + \text{sim}(F_0, F_2 - F_d))$$

$$= 0.5 \cdot (\text{sim}(F_0, F_1) + \text{sim}(F_0, F_d) + \text{sim}(F_0, F_2) - \text{sim}(F_0, F_d))$$

$$= 0.5 \cdot (\text{sim}(F_0, F_1) + \text{sim}(F_0, F_2))$$

$$= 0.5 \cdot (S_1 + S_2)$$

$$= S$$

[0036] The above calculations prove that the average similarity score S^e obtained based on the encrypted feature vectors E_1 and E_2 and the final similarity score S are identical. Effectively, the face recognition system can perform verification of a face to be authenticated based on E_1 and E_2 , which are random vectors that do not contain information about the original reference faces. One can also appreciate that the original reference faces cannot be recovered from E_1 and E_2 and thus privacy protection can be achieved.

[0037] In some implementations, more than two reference faces can be used and the encryptor 202 can be configured to encrypt more than two reference face features. In some implementations, the encryptor 202 can be configured to encrypt F_1 and F_2 using other known encryption techniques (such as other feature value confusion methods or feature length confusion methods) while allowing the final similarity score S to be calculated based on E_1 and E_2 . The implementations are not limited.

[0038] The present specification may further provide methods, systems, and apparatuses for performing face recognition with increased privacy protection, with two or more reference faces being processed by the methods described hereinabove. Figure 3 is a flow chart 300 of an example method for face recognition. The example method includes the following steps:

- 310: receiving an authentication input;
- 320: determining respective similarity scores between the authentication input and a plurality of encrypted features, wherein the plurality of encrypted features are obtained by encrypting each of a plurality of reference features that correspond to a plurality of reference faces of a user;
- 330: calculating, based on the respective similarity scores, a final similarity score between the authentication input and the plurality of reference faces of the user; and
- 340: verifying the authentication input based on the final similarity score.

[0039] At step 310, an authentication input is received. In some embodiments, the authentication input can be a face image of a user to be authenticated, for example, a user's face image collected by a device camera during a login procedure (such as unlocking a smartphone) or during a payment procedure. The received face image can be converted to a face feature vector by a face recognition model for subsequent calculations. Alternatively, the authentication input can be in the face feature vector form corresponding to a user's face data in other embodiments. The implementations are not limited.

[0040] At step 320, respective similarity scores between the authentication input and a plurality of encrypted features can be determined. In some embodiments, the respective similarity scores can be determined using one or more linear similarity functions, such as a vector dot product. The encrypted features can be obtained by encrypting each of a plurality of reference features that correspond to a plurality of reference faces of a user, for example, according to step 120 of the storing reference faces method in Figure 1. In implementations, the encrypted features can be pre-stored in the face recognition system. Advantageously, the face recognition system does not have the original reference faces or face feature vectors stored therein.

[0041] At step 330, a final similarity score between the authentication input and the plurality of reference faces of the user can be calculated based on the respective similarity scores determined at step 320. In implementations where

the respective similarity scores are determined using one or more linear similarity functions, the final similarity score can be calculated by calculating an average value of the respective similar scores.

[0042] At step 340, the authentication input can be verified based on the final similarity score. As described herein, verifying the identity of the authentication input can include different scenarios, such as verifying if the authentication input matches a particular user identity (e.g. face recognition based login for unlocking a personal smartphone or for eKYC) or matches one particular identity in a group of registered identities (e.g. face recognition based payment on merchant devices, face searching, face annotation, or photo album generation). For example, when a user uses face recognition based login to unlock a smartphone, the face recognition system can have pre-stored therein encrypted features corresponding to two or more reference faces of the smartphone owner. Upon calculation, if a final similarity score of the authentication input is low, the system may determine that the input may not be the rightful owner of the phone and the phone cannot be unlocked. In another example, in a face searching scenario, the face recognition system may have pre-stored therein encrypted features corresponding to reference faces of all the registered users. The system may then calculate a respective final similarity score associated with each of the registered users, and verify if the identity of the authentication input matches any of the registered users. The implementations of verifying the authentication input are not limited. In the scenarios described above, the face recognition system can be configured to output a similarity result for the authentication input, without having the original reference faces or the reference features of the users stored in the system.

[0043] One may appreciate that processing the reference faces as shown in Figure 1 and the face recognition as shown Figure 3 can be a combined process, in which references faces are processed by encryption and face recognition is performed subsequently. Alternatively, processing the reference faces and the face recognition can be two separate processes, performed by either the same party or different parties. Further, the face recognition method for can be implemented alone or in combination with other methods of identity authentication and identity proofing. The implementations are not limited.

[0044] Figure 4A is a schematic diagram of an example apparatus 400 including modules for processing reference faces. The apparatus 400 at least includes a

face recognition module 402, an encryption module 404, and a discard module 408. With reference to Figure 1 and Figure 2, the face recognition module 402 can be configured to generate a plurality of reference features corresponding to a plurality of reference faces of a user, such as converting the reference faces to reference face feature vectors. The encryption module 404 can be configured to encrypt each of the plurality of reference features to obtain a plurality of encrypted features. In some embodiments, the apparatus 400 may include a random number generation module 406, configured to generate a random noise feature for encryption. The random number generation module 406 may be part of the encryption module 404, or a separate module that is configured to generate and transmit the random noise feature to the encryption module 404. The discard module 408 can be configured to discard the plurality of reference faces, the reference features, and/or the random noise feature. The apparatus 400 may further include a storing module 410, configured to store the plurality of encrypted features in the face recognition system.

[0045] Figure 4B is a schematic diagram of an example apparatus 450 including modules for face recognition. The apparatus 450 at least includes a receiving module 452, a determining module 454, a calculation module 456, and a verification module 458. With reference to Figure 3, the receiving module 452 can be configured to receive an authentication input, which can include receiving a face image to be authenticated during a login or payment procedure. The received face image may be converted to a face feature vector for subsequent processing. The determining module 454 can be configured to determine respective similarity scores between the authentication input and a plurality of encrypted features. The plurality of encrypted features can be obtained by encrypting each of a plurality of reference features corresponding to a plurality of reference faces of a user. The calculation module 456 can be configured to calculate a final similarity score between the authentication input and the plurality of reference faces of the user based on the respective similarity scores. In some embodiments, the respective similarity scores can be determined by one or more linear similarity functions, such as a vector dot product. The verification module 458 is configured to verify the authentication input based on the final similarity score calculated by the 456.

[0046] The system, apparatus, module, or unit illustrated in the previous embodiments can be implemented by using a computer chip or an entity, or can

be implemented by using a product having a certain function. A typical embodiment device is a computer (and the computer can be a personal computer), a laptop computer, a cellular phone, a camera phone, a smartphone, a personal digital assistant, a media player, a navigation device, an email receiving and sending device, a game console, a tablet computer, a wearable device, or any combination of these devices. The modules described as separate parts may or may not be physically separate, and parts displayed as modules may or may not be physical modules, may be located in one position, or may be distributed on a number of network modules. Some or all of the modules can be selected based on actual demands to achieve the objectives of the solutions of the specification. A person of ordinary skill in the art can understand and implement the embodiments of the present application without creative efforts.

[0047] Figure 5 is a block diagram of an example computer system 500 suitable for executing at least some steps of the example methods shown in Figures 1 to 3. The following description of the computer system / computing device 500 is provided by way of example only and is not intended to be limiting.

[0048] As shown in Fig. 5, the example computing device 500 includes a processor 502 for executing software routines. Although a single processor is shown for the sake of clarity, the computing device 500 may also include a multi-processor system. The processor 502 is connected to a communication infrastructure 506 for communication with other components of the computing device 500. The communication infrastructure 506 may include, for example, a communications bus, cross-bar, or network.

[0049] The computing device 500 further includes a main memory 504, such as a random access memory (RAM), and a secondary memory 510. The secondary memory 510 may include, for example, a storage drive 512, which may be a hard disk drive, a solid state drive or a hybrid drive and/or a removable storage drive 514, which may include a magnetic tape drive, an optical disk drive, a solid state storage drive (such as a USB flash drive, a flash memory device, a solid state drive or a memory card), or the like. The removable storage drive 514 reads from and/or writes to a removable storage medium 518 in a well-known manner. The removable storage medium 518 may include magnetic tape, optical disk, non-volatile memory storage medium, or the like, which is read by and written to by removable storage drive 514. As will be appreciated by persons skilled in the relevant art(s), the removable storage

medium 518 includes a computer readable storage medium having stored therein computer executable program code instructions and/or data.

[0050] In an alternative implementation, the secondary memory 510 may additionally or alternatively include other similar means for allowing computer programs or other instructions to be loaded into the computing device 500. Such means can include, for example, a removable storage unit 522 and an interface 520. Examples of a removable storage unit 522 and interface 520 include a program cartridge and cartridge interface (such as that found in video game console devices), a removable memory chip (such as an EPROM or PROM) and associated socket, a removable solid state storage drive (such as a USB flash drive, a flash memory device, a solid state drive or a memory card), and other removable storage units 522 and interfaces 520 which allow software and data to be transferred from the removable storage unit 522 to the computer system 500.

[0051] The computing device 500 also includes at least one communication interface 524. The communication interface 524 allows software and data to be transferred between computing device 500 and external devices via a communication path 526. In various embodiments of the specification, the communication interface 524 permits data to be transferred between the computing device 500 and a data communication network, such as a public data or private data communication network. The communication interface 524 may be used to exchange data between different computing devices 500 which such computing devices 500 form part an interconnected computer network. Examples of a communication interface 524 can include a modem, a network interface (such as an Ethernet card), a communication port (such as a serial, parallel, printer, GPIB, IEEE 1394, RJ45, USB), an antenna with associated circuitry and the like. The communication interface 524 may be wired or may be wireless. Software and data transferred via the communication interface 524 are in the form of signals which can be electronic, electromagnetic, optical or other signals capable of being received by communication interface 524. These signals are provided to the communication interface via the communication path 526.

[0052] As shown in Fig. 5, the computing device 500 further includes a display interface 528 which performs operations for rendering images to an associated display 530 and an audio interface 532 for performing operations for playing audio content via associated speaker(s) 534.

[0053] As used herein, the term "computer program product" may refer, in part, to removable storage medium 518, removable storage unit 522, a hard disk installed in storage drive 512, or a carrier wave carrying software over communication path 526 (wireless link or cable) to communication interface 524. Computer readable storage media refers to any non-transitory, non-volatile tangible storage medium that provides recorded instructions and/or data to the computing device 500 for execution and/or processing. Examples of such storage media include magnetic tape, CD-ROM, DVD, Blu-ray™ Disc, a hard disk drive, a ROM or integrated circuit, a solid state storage drive (such as a USB flash drive, a flash memory device, a solid state drive or a memory card), a hybrid drive, 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 computing device 500. 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 computing device 500 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.

[0054] The computer programs (also called computer program code) are stored in main memory 504 and/or secondary memory 510. Computer programs can also be received via the communication interface 524. Such computer programs, when executed, enable the computing device 500 to perform one or more features of embodiments discussed herein. In various embodiments, the computer programs, when executed, enable the processor 607 to perform features of the above-described embodiments. Accordingly, such computer programs represent controllers of the computer system 500.

[0055] Software may be stored in a computer program product and loaded into the computing device 500 using the removable storage drive 514, the storage drive 512, or the interface 520. The computer program product may be a non-transitory computer readable medium. Alternatively, the computer program product may be downloaded to the computer system 500 over the communication path 526. The software, when executed by the processor 502, causes the computing device 500 to perform the necessary operations to execute the method as shown in Figures 1 and 2.

[0056] It is to be understood that the embodiment of Fig. 5 is presented merely by way of example to explain the operation and structure of the system 500. Therefore, in some embodiments one or more features of the computing device 500 may be omitted. Also, in some embodiments, one or more features of the computing device 500 may be combined together. Additionally, in some embodiments, one or more features of the computing device 500 may be split into one or more component parts.

[0057] It will be appreciated that the elements illustrated in Fig. 5 function to provide means for performing the various functions and operations of the system as described in the above embodiments.

[0058] It will be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present specification 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.

CLAIMS

What is claimed is:

1. A computer-implemented method for processing reference faces, comprising:
generating a plurality of reference features respectively corresponding to a plurality of reference faces of a user, wherein the plurality of reference faces of the user comprise at least a first reference face of the user and a second reference face of the user;
encrypting each of the plurality of reference features using a random value to obtain a plurality of encrypted features, wherein the plurality of encrypted features are associated with the plurality of reference features such that a final similarity score between an authentication input and the plurality of reference faces of the user can be calculated based on the authentication input and the plurality of encrypted features; and
discarding the plurality of reference features and the random value, such that the plurality of reference features are not recoverable from the plurality of encrypted features.
2. The method as claimed in claim 1, further comprising storing the plurality of encrypted features.
3. The method as claimed in claim 1 or 2, wherein the plurality of reference features comprise reference face feature vectors, and wherein generating the plurality of reference features corresponding to the plurality of reference faces of the user comprises:
receiving the plurality of reference faces of the user;
converting, by a face recognition model, each of the plurality of reference faces to a corresponding reference face feature vector; and
discarding the plurality of reference faces.
4. The method as claimed in claim 3, wherein encrypting each of the plurality of reference features comprises:
generating, by a random number generator, a random noise feature vector as the random value; and
encrypting each of the reference face feature vectors based on the random noise feature vector.
5. The method as claimed in claim 4, wherein the first reference face is converted to a first reference face feature vector and the second reference face is converted to a second reference face feature vector, and wherein encrypting each of the reference face feature vectors comprises:

encrypting the first reference face feature vector by adding the random noise feature vector to the first reference face feature vector; and

encrypting the second reference face feature vector by subtracting the random noise feature vector from the second reference face feature vector.

6. A computer-implemented method for face recognition, comprising:
 - receiving an authentication input;
 - determining respective similarity scores between the authentication input and a plurality of encrypted features, wherein the plurality of encrypted features are obtained by encrypting, using a random value, each of a plurality of reference features that respectively correspond to a plurality of reference faces of a user, wherein the plurality of reference faces of the user comprise at least a first reference face of the user and a second reference face of the user, and wherein the plurality of reference features are not recoverable from the plurality of encrypted features without the random value;
 - calculating, based on the respective similarity scores, a final similarity score between the authentication input and the plurality of reference faces of the user; and
 - verifying the authentication input based on the final similarity score.
7. The method as claimed in claim 6, wherein determining the respective similarity scores comprises determining the respective similarity scores using a linear similarity function.
8. The method as claimed in claim 7, wherein the linear similarity function comprises a vector dot product.
9. The method as claimed in claim 7 or 8, wherein calculating, based on the respective similarity scores, the final similarity score between the authentication input and the plurality of reference faces of the user comprises calculating an average value of the respective similarity scores.
10. The method as claimed in any one of claims 6 to 9, wherein receiving the authentication input comprises:
 - receiving a face image of a user to be authenticated; and
 - converting, by a face recognition model, the received face image to a face feature vector.

11. The method as claimed in 10, wherein the face image comprises a face image collected during a login procedure or during a payment procedure.
12. A computer-implemented system, comprising:
 - one or more processors; and
 - one or more computer-readable memories coupled to the one or more processors and having instructions stored thereon that are executable by the one or more processors to perform the method of any one of claims 1 to 11.
13. An apparatus comprising a plurality of modules for performing the method of any one of claims 1 to 11.