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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 September 2015 (11.09.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/132016 A1

(51) International Patent Classification:
G06K 9/00 (2006.01) *G06K 9/62* (2006.01)

(21) International Application Number:
PCT/EP2015/051169

(22) International Filing Date:
21 January 2015 (21.01.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
14305336.1 7 March 2014 (07.03.2014) EP

(71) Applicant: **THOMSON LICENSING** [FR/FR]; 1-5 rue Jeanne d'Arc, F-92130 Issy-les-Moulineaux (FR).

(72) Inventors: **SHARMA, Gaurav**; Technicolor R&d France, 975 Avenue Des Champs Blancs, F-35576 Cesson Sevigne Cedex (FR). **PEREZ, Patrick**; Technicolor R&d France, 975 Avenue Des Champs Blancs, F-35576 Cesson Sevigne Cedex (FR).

(74) Agents: **HUCHET, Anne** et al.; Technicolor, 1-5 rue Jeanne d'Arc, F-92130 Issy-Les-Moulineaux (FR).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,

[Continued on next page]

(54) Title: DEVICES AND METHODS FOR LEARNING AND APPLYING A DISTANCE METRIC

(57) Abstract: A distance metric (63) is learned with a training set (61) including training images and pairwise information indicating whether subsets of those images are pairwise corresponding to same objects. The distance metric is associated (41) with adjustable projection spaces, adapted to projecting the training images into those projection spaces and to computing related local distances. The projection spaces are selected (42) in function of pairs of the subsets of training images, the distance metric corresponding to the selected spaces. Adjusted projection spaces are accordingly determined (43, 44) by reducing offsets between expected values based on the pairwise information and effective values, of the distance metric with respect to the objects for the training images. A distance metric is also applied to study images, thereby deciding whether those images correspond pairwise to same objects. Those are relevant to face verification and to identity based clustering of faces.

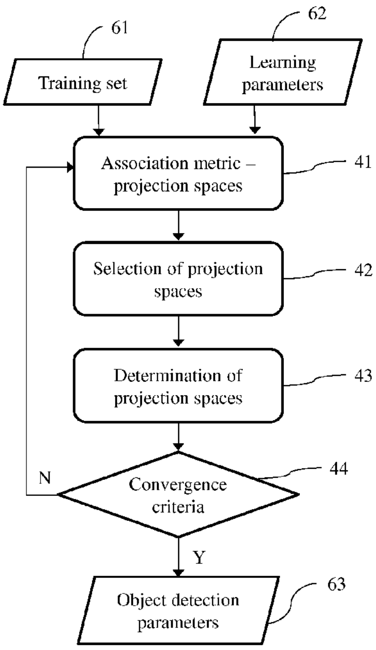

Fig 4

WO 2015/132016 A1

SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

— *with international search report (Art. 21(3))*

DEVICES AND METHODS FOR LEARNING AND APPLYING A DISTANCE METRIC

1. Technical field

The present disclosure relates to the domain of automatic analysis of objects, and more especially of faces, notably for face verification and for identity based clustering of faces. It pertains to the learning as well as to the application of corresponding distance metrics.

2. Background art

Automatic analysis of objects, more especially of faces, in digital images and videos is a very important problem in computer vision. It has many applications in recognizing, searching, retrieving and indexing images, for e.g.:

- Surveillance and video archives – find a person in large amounts of videos,
- Security – allow access to a person, or not, to a resource and
- Consumer databases – find a certain person in private or online image databases, such as those exploited on the market under the trademarks Flickr or Facebook.

In particular, face verification, i.e. determining if two faces are of the same person or not, has emerged during the last years as an important research problem. This is typically implemented by exploiting a system relying on a supervised learning framework, through using an annotated set containing (i) pairs of face images of the same person (taken at different times and conditions) and (ii) pairs of face images of different persons. That set is provided to the system, which has to predict whether a new pair (of unseen faces/person(s)) is of the same person or not.

Another field of applications of such a supervised learning framework consists in identity based clustering of faces.

Similar actions of verification or identity based clustering prove particularly interesting for spatio-temporal face tracks, also called face tubes. The task with realistic videos is much more challenging than with still image datasets. Indeed, for the latter, the faces are usually near frontal with good illumination, while for the former, the faces have unconstrained pose and expression variations, and are further taken in diverse and sometimes difficult illumination conditions. That leads to failure cases of existing systems in a number of situations, where the faces have challenging poses, expressions and illuminations.

In article “Unsupervised Metric Learning for Face Identification in TV Video”, ICCV 2011 – International Conference on Computer Vision, Nov. 2011, Barcelona, Spain, *IEEE*, pp. 1559-1566, R. G. Cinbis, J. Verbeek and C. Schmid dealt with face tube matching for face identification. They relied on a cast-specific learned distance metric derived from a distance metric learned from static face pairs. They showed that such a cast-specific metric performs better than a generic metric learned from external data, and applied it to unconstrained consumer videos. They further tested distances between tracks by comparing min-min distances over the faces in each track on one hand, with average face-to-face distances based on all possible face pairs from the two respective tracks, on the other hand.

Though providing relatively satisfying results with respect to other existing processes, such a system was carried out in the above article in relation with frontal face detectors providing the evaluation dataset, being thereby limited to near frontal poses. As a matter of fact, the need for enhanced precision remains quite strong, especially under challenging variations in pose, expression and illumination conditions.

3. Summary

The purpose of the present disclosure is to provide solutions enabling potentially a good degree of accuracy in object identification, in particular in face identification, with respect to existing systems.

More precisely, the present disclosure aims notably at providing potentially efficient systems for face verification or face clustering, as regards still images as well as face tubes.

Among others, an object of the present disclosure is a face tube verification system making possible, in its best embodiments, enhanced precision with respect to state-of-the-art solutions, including under challenging poses, expressions and illuminations.

In this respect, the present disclosure relates to a device adapted for learning a distance metric with at least one set of training images representative of at least two objects and with associated pairwise information indicating whether subsets of those training images are pairwise corresponding to same of those objects. According to the present disclosure, the device comprises at least one processor configured for:

- associating the distance metric with adjustable projection spaces, adapted to projecting the training images into any of those adjustable projection spaces and to computing local distances in the adjustable projection space corresponding to those projected training images,

- obtaining the distance metric for the training images as those local distances by selecting, in function of pairs of the subsets of the training images, the adjustable projection spaces corresponding respectively to the distance metric applied to those pairs of the subsets,

- determining adjusted projection spaces from respectively those adjustable projection spaces by reducing offsets between expected values based on that pairwise information and effective values, of the distance metric with respect to the objects for the training images.

The number of the adjustable projection spaces is advantageously predefined.

In particular embodiments, the set(s) of training images comprising training tubes of images, the processor(s) is/are configured for:

- projecting those training tubes into the adjustable projection spaces,

- computing the local distances between those projected training tubes,

- obtaining the distance metric for each pair of the training tubes by selecting one of the adjustable projection spaces in function of the local distances between the projected training tubes of that pair.

Advantageously, the processor(s) is/are then configured for computing the local distances between two of the projected training tubes by:

- selecting two projected training images of respectively the training tubes in function of local distances between the projected training images of the training tubes,

- computing the local distances between those selected projected training images.

Also, the processor(s) is/are advantageously configured for obtaining the distance metric for any pair of the training tubes by minimizing the local distances between the projected training images belonging to the training tubes over the projection spaces.

In other particular embodiments, the processor(s) is/are configured for:

- computing the local distances between the projected training images,
- obtaining the distance metric for each pair of the training images by selecting one of the projection spaces in function of the local distances between the respective projected training images of that pair.

Preferably, the processor(s) is/are configured for determining the adjusted projection spaces by proceeding with successive iterations in applying modifications to the adjustable projection spaces, based respectively on successive groups of sample images extracted from the training images.

In specific implementations, the processor(s) is/are then configured for implementing a minimization of a hinge loss based on those groups of sample images, involving the distance metric applied to those groups of sample images and the effective values of the distance metric with respect to the objects for the sample images.

In advantageous embodiments, the processor(s) is/are configured for:

- representing the training images by training feature elements, each having a determined number of feature dimensions,
- determining a basis having a limited number of basis dimensions with respect to that determined number of feature dimensions, founded on those training feature elements,
- projecting those training feature elements onto that basis having that limited number of basis dimensions.

Preferably, then, the processor(s) is/are configured for initializing each of the projection spaces within that basis having that limited number of basis dimensions, by means of a respective limited number of those training images.

A further object of the present disclosure is a device adapted for applying a distance metric to at least one set of study images representative of at least two objects. That device is configured for applying pairwise that distance metric to subsets of those study images, for comparing resulting values to at least one threshold, and for deciding that those subsets of the study images correspond pairwise to a same of those objects if those resulting values are lower than that threshold.

According to the present disclosure, the device comprises at least one processor configured for:

- associating the distance metric with projection spaces, adapted to projecting those study images into any of those projection spaces and to computing local distances in the projection space corresponding to those projected study images,

- obtaining the distance metric for those study images as those local distances by selecting, in function of pairs of those subsets of the study images, the projection space corresponding respectively to the distance metric applied to those pairs of the subsets.

The device adapted for applying a distance metric is preferably configured for exploiting the distance metric obtained by the device adapted for learning a distance metric according to any embodiments of the present disclosure.

The disclosure further pertains to a face detection system including a device adapted for learning a distance metric and/or a device adapted for applying a distance metric, according to any of their embodiments.

In addition, the present disclosure concerns a method for learning a distance metric with at least one set of training images representative of at least two objects and with associated pairwise information indicating whether subsets of those training images are pairwise corresponding to same of those objects.

According to the disclosure, the method comprises:

- associating the distance metric with adjustable projection spaces, adapted to projecting those training images into any of those adjustable projection spaces and to computing local distances in the adjustable projection space corresponding to those projected training images,

- obtaining the distance metric for the training images as those local distances by selecting, in function of pairs of those subsets of the training images, the adjustable projection space corresponding respectively to the distance metric applied to those pairs of the subsets,

- determining adjusted projection spaces from respectively the adjustable projection spaces by reducing offsets between expected values based on that pairwise information and effective values, of the distance metric with respect to the objects for the training images.

The method for learning a distance metric is preferably implemented by a device for learning a distance metric according to any embodiments of the present disclosure.

The disclosure also concerns a method for applying a distance metric to at least one set of study images representative of at least two objects. That method includes applying pairwise the distance metric to subsets of those study images, comparing resulting values to at least one threshold, and deciding that those subsets of the study images correspond pairwise to a same of those objects if those resulting values are lower than that threshold.

According to the present disclosure, the method comprises:

- associating the distance metric with projection spaces, adapted to projecting those study images into any of those projection spaces and to computing local distances in the projection space corresponding to those projected study images,

- obtaining the distance metric for the study images as those local distances by selecting, in function of those subsets of the study images, the projection spaces corresponding respectively to the distance metric applied to those pairs of the subsets.

The method for applying a distance metric is preferably implemented by a device for applying a distance metric according to any embodiments of the present disclosure.

The disclosure also relates to a computer program comprising software code adapted to perform steps of a method for learning a distance metric or of a method for applying a distance metric, compliant with any execution modes of the present disclosure.

In addition, the present disclosure pertains to a method of comparing at least a first image representative of a first object with at least a second image representative of a second objet, the method comprising:

- projecting the first and second images in a plurality of projection spaces, each projection space being representative of two different variations,

- for each projection space, comparing the result of the projection of the first image with the result of the projection of the second image,

- selecting the projection space for which the comparison result is minimal.

According to particular implementations, the method further comprises comparing a difference between the first image projected in the selected space and the second image projected in the selected space with a threshold value.

In more specific implementations, the method further comprises classifying the first and second objects according to the comparison result with the threshold value. Advantageously, if that difference is less than the threshold value then the first and second objects are classified in a same category and if that difference is greater than the threshold value then the first and second objects are classified in two different categories.

Preferably, those variations belong to a group of variations comprising pose variation, illumination variation, contrast variation, expression variation, age variation, accessories variation, tone variation.

The disclosure further relates to a device configured for comparing at least a first image representative of a first object with at least a second image representative of a second object, the device comprising:

- means for projecting the first and second images in a plurality of projection spaces, each projection space being representative of two different variations,

- for each projection space, means for comparing the result of the projection of the first image with the result of the projection of the second image,

- means for selecting the projection space for which the comparison result is minimal.

According to particular embodiments, the device further comprises means for comparing a difference between the first image projected in the selected space and the second image projected in the selected space with a threshold value.

In more specific implementations, the device further comprises means for classifying the first and second objects according to the comparison result with the threshold value.

The disclosure concerns also a device configured for comparing at least a first image representative of a first object with at least a second image representative of a second object, the device comprising at least one processor configured for:

- projecting the first and second images in a plurality of projection spaces, each projection space being representative of two different variations,
- for each projection space, comparing the result of the projection of the first image with the result of the projection of the second image,
- selecting the projection space for which the comparison result is minimal.

Another object of the disclosure is a computer program product, comprising instructions of program code for executing the method of comparing first and second images in any of its implementations, when that program is executed on a computer.

An additional object of the disclosure is a processor readable medium having stored therein instructions for causing a processor to perform at least the method of comparing first and second images in any of its implementations.

4. List of figures

The present disclosure will be better understood, and other specific features and advantages will emerge upon reading the following description of particular and non-restrictive illustrative embodiments, the description making reference to the annexed drawings wherein:

- figure 1 is a block diagram representing schematically an object detection system, including devices adapted for learning and applying a distance metric according to the present disclosure;
- figure 2 shows more in detail the parts of the system of figure 1 adapted for learning a distance metric;
- figure 3 shows more in detail the parts of the system of figure 1 adapted for applying a distance metric;
- figure 4 is a flow chart illustrating steps executed for learning a distance metric with the system of figure 1;

- figure 5 is a flow chart illustrating steps executed for applying a distance metric with the system of figure 1;
- figure 6 diagrammatically shows a metric learning apparatus comprising the metric learning features of the system represented on figure 1.

5. Detailed description of embodiments

The present disclosure will be described in reference to a particular functional embodiment of an object detection system 1, as illustrated on **Figure 1**. The object detection system 1 is adapted for detecting similarities and dissimilarities of objects appearing on images, and for outputting corresponding information. Particularly advantageous implementations include object verification, object clustering, or a combination of both. In preferred embodiments, such images are faces, which makes the object detection system 1 appropriate for face verification and/or identity based clustering of faces.

Preferably, the system 1 is designed for processing videos, which in appropriate implementations can provide significantly enhanced detection capabilities with respect to mere comparisons of still images. For faces, this amounts to taking face tubes into consideration. In other implementations, the system 1 is directed to still images. Also, in more sophisticated embodiments, system 1 is provided with both capacities.

The system 1 includes units 2 and 3 for respectively learning and applying distance metrics, each of the learning unit 2 and application unit 3 comprising several modules. The distance metrics are learned by means of the learning unit 2 on the ground of at least one set of training images representative of the considered objects, and of associated pairwise information indicating whether subsets of the training images correspond pairwise to the same objects or not. Hereinbelow, the terms “training set” refer to the training images together with the associated pairwise information.

Also, the thereby obtained distance metrics are applied by means of the application unit 3 to study images, in which possibly similar objects can be identified. The application unit 3 is able to provide the appropriate output information pertaining to object verification and/or clustering, namely whether those study images correspond to a same of those objects or not.

The comparison is pairwise applied to the subsets of images, that comparison between any two of those subsets leading to a conclusion that they are either corresponding to a same of the represented objects or not.

For the detection system being directed to still images, each subset preferably consists in a unique image, so that two images are compared. By contrast, for the detection system being directed to videos, each subset is preferably representative of a spatio-temporal object track (i.e. a face tube for objects consisting in faces). Then, in a particular embodiment, each subset of the images is made of all the frames over which the object track extends – which number depends generally on the considered track. In an alternative embodiment, each subset of the images is made of part only of the frames over which the object track extends, which can notably consist in a given predefined number or be truncated to a maximum number.

Also, the comparison is advantageously effected on data extracted from the images and representative of the objects therein – typically image vectors. Preferably, parts only of those data are then exploited in the comparisons.

In particular implementations, the system 1 is an apparatus, or a physical part of an apparatus, designed, configured and/or adapted for performing the mentioned functions and produce the mentioned effects or results. In specific implementations, the system 1 is constituted by a CPU, memory resources and user interface capacities.

In alternative implementations, the system 1 is embodied as a set of apparatus or physical parts of apparatus, whether grouped in a same machine or in different, possibly remote, machines. Preferably, then, the unit 2 for learning distance metrics and the unit 3 for applying distance metrics are implemented in different apparatus.

The units 2 and 3 and associated modules are to be understood as functional entities rather than material, physically distinct, components. They can consequently be embodied either as grouped together in a same tangible and concrete component, or distributed into several such components. Also, each of those units 2 and 3 and modules is possibly itself shared between at least two physical components.

In addition, units 2 and 3 and associated modules are implemented in hardware, software, firmware, or any mixed form thereof as well. They are preferably embodied within at least one processor of the system 1.

The terms “adapted” and “configured” are further used in the present disclosure as broadly encompassing initial configuration, later adaptation or complementation of the present device, or any combination thereof alike, whether effected through material or software means (including firmware).

The system 1 also comprises one or several user interface(s) 11. Those user interfaces 11 are appropriate for any communications between a user and the system 1 in either direction, whether notably for entering data or selections by the user, or for providing the user with information, menus or warnings by the system 1. Depending on the implementations, the communications encompass in particular visual, verbal or tactile signals, or any combinations thereof.

The system 1 further includes two storage units 12 and 13, adapted for storing data and information relevant to the exploitation of the respective learning and application units 2 and 3. Means are provided for making some output data obtained in the storage unit 12 from the learning unit 2 available to the application unit 3 in the storage unit 13 – this being represented on Figure 1 by dashed lines between storage units 12 and 13.

The storage units 12 and 13 include preferably one or several ROM (read-only memory), notably for program instructions corresponding to the learning and application units 2 and 3, and dynamic memory resources such as e.g. those based on RAM (Random Access Memory) or EEPROM (Electrically-Erasable Programmable Read-Only Memory) capacities. In advantageous embodiments, the latter include at least one Flash memory, possibly within one or several SSD (Solid-State Disk).

Though represented functionally as separate entities on the drawings so as to illustrate the associations with the respective learning and application units 2 and 3, the storage units 12 and 13 can take various material forms. In particular, in some embodiments in which the learning and application units 2 and 3 are implemented in a same apparatus, the storage units 12 and 13 are given by same component(s). In other embodiments, parts of units 12 and 13 are constituted by a same material entity, used for recording outputs from learning unit 2 and for making them available to the application unit 3, while remaining parts of storage units 12 and 13 are made of distinct components. In still alternative implementations, corresponding notably to distinct apparatus for respectively the learning unit 2 and application unit 3, storage units 12 and 13 are physically separated. In the latter case, transmission means are provided for retrieving appropriate data in storage unit 13 from storage unit 12.

The storage unit 12 is adapted to store input and output data associated with the learning unit 2, which typically includes one or several training sets as well as learning control parameters, for the inputs, and object detection parameters yielded by the training unit 2, for the outputs. The storage unit 13 is adapted to store input and output data associated with the application unit 3, which typically includes the object detection parameters produced by unit 2 together with user-defined detection control parameters, as well as with the study images, for the inputs, and results pertaining to objects correspondence in the study images, for the outputs.

As shown on **Figure 2**, the learning unit 2 is cooperating with user interface 11 and storage unit 12, and it includes an association module 21, a selection module 22 and a determination module 23.

The association module 21 is configured for associating a distance metric under determination with adjustable projection spaces. Each of those projection spaces is adapted to the projection of the training images into it, and to the computation of local distances corresponding to the training images so projected. The projection spaces are adjustable insofar as each of them can be iteratively adjusted until a global convergence is deemed met for the whole distance metric, involving the whole set of associated adjustable projection spaces.

The selection module 22 is configured for obtaining the distance metric for the training images, by selecting the appropriate adjustable projection space corresponding to any pair of the image subsets. The distance metric is more precisely derived from the local distances in the projection spaces.

Preferably, the selection of the appropriate projection space for a given pair is obtained by a minimization process over the projection spaces, applied to the local distances between the two subsets of that pair as projected. Then, the distance metric applied to that pair is given by the local distance in the selected space, as will be developed below in particular implementations.

The determination module 23 is configured for determining adjusted projection spaces from the adjustable projection spaces. Namely, the projection spaces can be modified until a satisfying level of precision is globally obtained for the whole set of them. That adjustment of the projection spaces is based on reducing the offsets between expected values and effective values of the distance metric for the training images with respect to the considered objects. Indeed, as mentioned above, the distance metric is dynamically depending on the adjustable projection spaces with which it is associated. Also, expected values can be computed based on the pairwise information available in the training set(s).

Preferably, the offsets reduction is relying on a minimization process, parameterized through one or several convergence thresholds.

Since the adjustable projection spaces can be iteratively modified by the determination module 23, the association module 21 and the selection module 22 are led to renew their respective operations in transformed conditions – which is represented on Figure 2 by a return line from the determination module 23 to the association module 21.

The learning unit 2 is thus able to produce appropriate data corresponding to the selected projection spaces, on the ground of the training sets and of input parameters, such as e.g. the number of projection spaces, selected algorithms and initialization information, further to convergence threshold data.

It is interesting to note that the thereby obtained distance metric is flexibly derived from training sets, and is thus closely adapted to specificities of the considered objects and images.

Also, it deserves observing that several convergence variables exploited in the process are latent or hidden (i.e. not directly observed but inferred through the exploited models). This is particularly true for the selected projection spaces in relation with the corresponding images of the subsets.

As illustrated on **Figure 3**, the application unit 3 is cooperating with user interface 11 and storage unit 13, and it comprises an association module 31, a selection module 32 and a comparison module 33.

The association module 31 is configured for exploiting one or several distance metrics available from the storage unit 13, which are relying on projection spaces. The associated data are preferably known partly from the outputs of the learning unit 2, and partly from parameters entered by the user or automatically determined by default. In variant implementations, at least parts of the data are randomly set.

The association module 31 is adapted for associating the available distance metrics with the projection spaces, more precisely through projecting the study images retrieved from the storage unit 13 into any of the projection spaces and through computing related local distances between subsets in the projection space corresponding to the projected study images.

The selection module 32 is configured for obtaining the distance metric for the study images, by selecting the appropriate adjustable projection space corresponding to any pair of the image subsets. More specifically, the distance metric is given by the local distances between the subsets in the projection spaces.

The process for selecting the appropriate projection space is preferably identical to the process exploited in the learning unit 2 for the training images.

The comparison module 33 is adapted to applying the distance metric to any pair of the subsets of images, as stated from the association module 31 and selection module 32, to comparing the resulting value with a similarity threshold available from the storage unit 13, and to deciding that those two subsets are corresponding to a same object if and only if the resulting value is smaller (or alternatively: not greater) than that similarity threshold.

More precisely, the comparison module 33 is advantageously configured for executing a process directed to picking up the subsets of study images to be compared – successively and/or in parallel. Such a process involves algorithms suited to the kind of desired information, which typically relies on identifying the images displaying an object (or several ones) similar to a reference one (or several ones), or on clustering the study images into groups respectively displaying similar objects.

In a variant implementation, the application unit 3 is adapted to be fed by object detection data that are not produced at all by the learning unit 2, but are obtained otherwise. These can be notably generated by alternative processes, consist in preset tables of values and/or rely on user-entered information.

In operation, as shown for a preferred implementation on **Figure 4**, the learning unit 2 fed with a training set 61 and with learning parameters 62 executes the following steps:

- association of the distance metric with the projection spaces in step 41;

- selection of the projection spaces associated with the pairs of training image subsets in step 42;

- determination of the projection spaces in the frame of an iteration process in step 43, that iteration process being directed to reducing offsets between expected and effected values for the distance metric applied to the subsets of training images;

- consideration of convergence criteria for that iteration process in step 44, on the ground of one or several offset thresholds;

- return to the association step 41 with modified projection spaces if the convergence criteria are not met;

- end of the iteration process if the convergence criteria are met, and outputting appropriate object detection parameters 63, including data pertaining to the projection spaces obtained further to the iteration steps.

In operation, as shown for a preferred implementation on **Figure 5**, the application unit 3 is fed with object detection data 64, notably based advantageously on the object detection parameters 63 yielded by the training process of Figure 4, and study images 65. It executes the following steps:

- association of the distance metric with the projection spaces in step 51;

- selection of the projection spaces associated with the pairs of study image subsets in step 52;

- comparison of images in step 53, through application of the distance metric to subsets of the study images; that leads to outputting appropriate correspondence information 66 pertaining to those images (based on similarities / dissimilarities).

In preferred implementations, the learning unit 2 and/or application unit 3 are available through one or several computer programs, comprising software code adapted to execute the steps above (represented on Figures 4 and 5).

The computer program(s) can have any form, and notably be embedded in an apparatus for learning and/or for applying a distance metric. In alternative embodiments, they are available separately, and configured to be implemented within an apparatus so as to allow the execution of a process for learning and/or for applying a distance metric compliant with the present disclosure. This can be done either via a tangible support carrying the computer program(s) or via local or remote downloading. In particular, the computer program(s) can be available as flashing firmware, enabling to update an apparatus for learning and/or for applying a distance metric.

In still other embodiments, the computer program(s) is/are configured to be exploited in combination with an apparatus, but remotely, notably through online operations.

In particular implementations, a non-transitory program storage device is exploited, readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method for learning and/or applying a distance metric compliant with the present disclosure.

Such a non-transitory program storage device can be, without limitation, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor device, or any suitable combination of the foregoing. It is to be appreciated that the following, while providing more specific examples, is merely an illustrative and not exhaustive listing as readily appreciated by one of ordinary skill in the art: a portable computer diskette, a hard disk, a ROM, an EPROM (Erasable Programmable ROM) or a Flash memory, a portable CD-ROM (Compact-Disc ROM).

The object detection system 1 will be detailed below on some particular implementations, related to face verification with video face tubes. Those can prove particularly advantageous in best achievements, including under challenging variations in pose, expression and illumination conditions.

In the considered implementations of a metric learning framework – corresponding to the learning unit 2, such variations are preferably incorporated implicitly in a distance metric as latent variables and a stochastic gradient descent based (SGD) algorithm is preferably exploited. This leads to a latent max-margin metric learning, able to possibly provide significant performance improvements over usual technologies in multiple cases.

Spatio-temporal face tubes of a processed training set 61 may be obtained by using face detection and/or tracking technologies. A distance function is learned through the learning unit 2 for comparing such tubes through the application unit 3.

The tubes are denoted as $\bar{\mathbf{s}} = (\mathbf{s}, \mathbf{x}_s, v_s, f_s^0, N_s)$, where:

- v_s is a unique identifier for a video,

- f_s^0 is the frame in the video v_s where the tube begins,

- N_s is the number of frames over which the tube extends,

- $\mathbf{x}_s \in \mathbb{R}^{4\,x\,N_s}$ contains bounding boxes (x_1, y_1, x_2, y_2) for the N_s frames, and

- $\mathbf{s} \in \mathbb{R}^{d\,x\,N_s}$ contains some *d* dimensional feature vectors (namely vectors of numerical features that represent some objects, here corresponding to face descriptors) for all the faces in the tube, e.g. based on statistics of local pixel patterns – as familiar to persons skilled in the art.

The matrix $\boldsymbol{s}$ comprises N_s columns representing respectively feature vectors, the feature vector for the ith frame in tube $\bar{\mathbf{s}}$ being noted $\boldsymbol{s_i}$. Also, for tubes $\bar{\mathbf{s}}$ and $\bar{\mathbf{t}}$, a binary indicator $y_{st} \in \{-1, 1\}$ indicates if those tubes correspond to a same person ($y_{st} = 1$) or not ($y_{st} = -1$).

In particular embodiments, the bounding boxes are expanded to make them square and/or padded with black pixels when they go out of the image.

Any two face tubes $\bar{\mathbf{s}}$ and $\bar{\mathbf{t}}$ are compared by means of a distance function *D* (distance metric) defined as:

$$D^2(\mathbf{s}, \mathbf{t}) = \min_{l,p,q} \left\| L_l(\mathbf{s}_p - \mathbf{t}_q) \right\|^2 \qquad (1)$$

where:

- $(l, p, q) \in \mathbb{N}^3$ are latent variables;

- L_l (l = 1,…, k) is a projection matrix corresponding to a typical combination of facial variations, k being a predetermined number of projection matrices; more precisely, L_l is a lower triangle matrix providing a Mahalanobis like metric relying on the symmetric positive semi-definite matrix M_l given by $M_l = L_l^T L_l$ (with "*T*" standing for a matrix transposition);

- *p*, *q* specify frames in the respective face tubes $\bar{\mathbf{s}}$ and $\bar{\mathbf{t}}$.

Each matrix L_l corresponds to a projection in a subspace where a certain combination of face-variations may be compared, e.g. frontal-frontal or frontal-profile. The intuition behind such distance is thus that it is desired to find the faces (in the two considered tubes) that are closest to each other in the subspace that is optimized for comparing such face pair. This contrasts with existing solutions in which a single projection matrix is used for faces with all types of variations, i.e. pose, expression, lighting, etc., leading to embedding different pose pairs into the same space for comparison.

The training set 61 comprises a set of same and a set of different pairs of face tubes, which may be created in particular either using annotations or using co-occurrence arguments, as known by a skilled person and specified notably in the article by Cinbis et al. cited above. The projection matrices $\{L_l\}$ are then learned by carrying out a minimization process on a hinge loss advantageously defined as:

$$\mathcal{L} = \sum_{(\bar{\mathbf{s}},\bar{\mathbf{t}})} \max\ (0, 1 - y_{st}\ (b - D^2(\mathbf{s},\mathbf{t})) \qquad (2)$$

with respect to $\{b, L_l / l = 1, \ldots, k\}$, with b being an adjustment offset.

The optimization process is preferably performed with a stochastic algorithm, using subgradients of the hinge loss $\mathcal{L}$ with respect to projection matrices L_l given by:

$$\nabla_{L_l} \mathcal{L} = \begin{cases} 0 & \text{if } y_{st}\big(b - D^2(\mathbf{s},\mathbf{t})\big) > 1 \\ 2 y_{st} L_l (\mathbf{s} - \mathbf{t})(\mathbf{s} - \mathbf{t})^T & \text{otherwise} \end{cases} \qquad (3)$$

where the latent variables are computed for the current pair of tubes ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$) by using the optimization formula (1). Namely, the appropriate projection space (variable l) and appropriate tube frames belonging to the respective tubes (variables p and q) are induced from the minimization process.

The exploited stochastic algorithm, providing an SGD based learning of projections for comparing face tubes, is advantageously defined as follows:

1: Input: Annotated training pairs $\mathcal{T} = \{\bar{\mathbf{s}}, \bar{\mathbf{t}}, y_{st}\}$, learning rate ($r$ – which corresponds to the step size) and number of stochastic updates (*n-iters*)
2: Initialize: $\{L_l / l = 1, \ldots, k\}$
3: **for** $i = 1,\ldots,$ *n-iters* **do**
4: Randomly sample a training tube pair $\{\bar{\mathbf{s}}, \bar{\mathbf{t}}\}$ from the training set $\mathcal{T}$
5: Randomly sample up to m image vectors from $\bar{\mathbf{s}}$ and $\bar{\mathbf{t}}$
6: Compute (l, p, q) over the sampled vectors using formula (1)
7: $L_l \leftarrow L_l - r\, \nabla_{L_l} \mathcal{L}$ (with the latter being given by formula (3)
8: **end for**
9: Output: $\{L_l / l = 1, \ldots, k\}$

According to that implementation, at each stochastic step further to randomly sampling an annotated tube pair, (up to) a fixed number m of images from each of the tubes is also sampled. This involves considering only parts, rather than all, of the tube frames, for each application of the minimization process given by formula (1). That execution of the stochastic updates can enable to generate a much larger number of training points through constructing many more training pairs (especially positive pairs that are usually scarce), to drive the algorithm to update all of the projection matrices L_l more often and to get smoother estimates of those matrices.

More details will be now given about particular embodiments of the learning unit 2.

As concerns the determination of the feature vectors, it is advantageously proceeded by using local binary pattern (LBP) as the base features (LBP being a type of feature used for classification in computer vision and based on a division of examined windows into cells, familiar to a person skilled in the art). According to particular implementations, the LBPs are extracted in 3x3 circular pixel neighborhoods (with the diagonal pixels bilinearly interpolated) and used as uniform patterns, i.e. as having at most two bitwise transitions from 0 to 1, or vice versa, when the bit pattern is seen as circular. The LBPs are extracted densely at every pixel at three scales with face image sizes of 120, 80 and 60 pixel square.

Spatial histograms of the LBP features are then made on the ground of cell size of 10x10 pixels and all the L1-normalized (i.e. normalized with the taxicab norm) cell histograms are concatenated for all the scales, so as to give a final vector of 14,396 dimensions for each face image. That number of dimensions corresponds to the cumulated numbers of cells in a face image for the respective sizes (i.e. 12^2, 8^2, 6^2), multiplied by the number of labels associated with each cell (i.e. 59, of which 58 correspond to uniform patterns having at most two bitwise transitions, and 1 corresponds to all non-uniforms patterns). Namely, $14,396 = (12^2 + 8^2 + 6^2) \times 59$.

The obtained vectors are then advantageously processed as follows, notably to adjust the number of dimensions to the desired level. The method carried out relies on a PCA procedure (i.e. Principal Component Analysis, exploited for converting a set of observations into a set of values of linearly uncorrelated variables called principal components, as well known by a skilled person). Namely, the local pattern histograms are projected onto their PCA basis. This proves to provide a strong scheme in appropriate implementations.

More specifically, as a baseline:

- the local pattern histograms are ℓ_1-normalized and subsequently element-wise square root (or ℓ_2-) normalized; this corresponds to a non-linear Helinger kernel map, for which the Euclidean distances between mapped vectors amount to the Bhattacharyya distance between the probability distributions represented by original distances;

- the obtained high-dimensional vectors are projected onto their low rank PCA basis, that rank being predetermined for each projection space; for example the number of projection subspaces is equal to 3 (L_1, L_2, L_3) and their rank is the same and is worth 35 – which amounts to a compression of more than 137 with respect to the 14,396 original dimension.

Preferably, the metric learning is triggered only once the dimension has been reduced using PCA with proper normalization. This can indeed potentially offer performance similar to executing the metric learning over the original high-dimensional vectors, while being faster. In an alternative implementation, the metric learning is effected before the PCA low-dimensional projection.

As regards initialization of the projection matrices L_l, a preferred method consists in randomly selecting for each projection matrix a small number of training vectors (e.g. 1500) and in initializing that projection matrix with the low rank whitened PCA matrix of those vectors. The whitening is carried out by dividing each of the PCA vectors with the square root of the corresponding eigenvalue. Such a whitening proves to improve significantly the performance in multiple applications. This can be explained by the fact that many of the smaller eigenvalue components are relatively discriminative.

In a variant initialization of the projection matrices, an unsupervised clustering of faces is done using k-means (i.e. by distributing the feature vectors into clusters having the nearest means, based on a partitioning of the feature space into Voronoi cells). The resulting clusters are used for initializing the projection matrices.

According to a variant method, rather than PCA, another compression method is exploited.

Also, in other implementations, the minimum value used in formula (1) is replaced with a minimum over variable l and with average values over p and q. Namely, the norm for each projection subspace L_l is determined through the average L_l-distance between all possible face pairs for the two compared tube, and the smallest resulting average L_l-distance is selected over the projection subspaces as the distance function D.

Embodiments of the application unit 3 correspond to the particular learning units 2 described above. Namely, the determined projection subspaces are exploited for projecting the face tubes of the study images 65 and for determining whether those tubes correspond to the same person or not. This is based on formula (1), and on a threshold indicating whether the distance between two tubes is sufficiently low for considering that they relate to the same person.

In alternative embodiments of the object detection system 1 pertaining to still images, the latent variables are reduced to variable *l* pertaining to the projection subspace. The distance function *D* exploited in the minimization process is then given, for feature vectors **s** and **t**, by:

$$D^2(\mathbf{s},\mathbf{t}) = \min_l \|L_l(\mathbf{s} - \mathbf{t})\|^2,$$

while learning formulas (2) and (3) are similar to the above. The learning stochastic algorithm described for video face tubes is further simplified, since the random sampling is limited to the training pairs of image vectors and only latent variable *l* needs to be computed at each iteration step.

Such implementation, applied to static image verification, can potentially lead to discovering automatically and exploiting fine details in different faces.

A particular apparatus 7, visible on **Figure 6**, is embodying the learning unit 2 described above. In this represented example, the application unit 3 is not included in the apparatus 7, so that the metric learning outputs (the object detection parameters 63) are expected to be made available to another apparatus for processing the study images 65. This can be done either through remote transmission – e.g. via cable or wireless – or by storage on a removable support.

The apparatus 7 corresponds for example to a personal computer (PC), a laptop, a tablet, a smartphone or a games console. It comprises the following elements, connected to each other by a bus 75 of addresses and data that also transports a clock signal:

- a microprocessor 71 (or CPU) ;
- a non-volatile memory of ROM type 76;
- a RAM (Random Access Memory) 77;

- one or several I/O (Input/Output) devices 74 such as for example a keyboard, a mouse, a joystick, a webcam; other modes for introduction of commands such as for example vocal recognition are also possible;
- a power supply 78 ; and
- a network unit 79, such as a radiofrequency, cell network or cable communication unit.

According to a variant, the power supply 78 is external to the apparatus 7.

It is noted that the word "register" used in the description of memories 76 and 77 designates in each of those memories a memory zone of any size, which can cover low capacity (a few binary data) as well as large capacity (enabling to store a whole program or all or part of information representative of data calculated or to be displayed).

When switched-on, the microprocessor 71 loads and executes the instructions of the program contained in the register 760 of the ROM 76.

The random access memory 77 notably comprises:

- in a register 771, the training set 61,
- in a register 772, the learning parameters 62;
- in a register 773, the face detection parameters corresponding to the object detection parameters 63.

An example of an apparatus embodying the application unit 3 is similar to apparatus 7, except that the program register 760 is loaded with a program directed to metric application, and that the registers 771 to 773 are replaced with registers configured for storing the study images 65, face detection data 64 and image correspondence outputs 66.

In a variant implementation illustration of the system 7, the apparatus 7 is completed so as to include in its ROM program register 760 the functionalities of both the learning unit 2 and the application unit 3, and to have its RAM registers adapted to store the training set 61, learning parameters 62 and face detection parameters 63 as well as the face detection data 64, study images 65 and image correspondence outputs 66.

According to a variant, the program for metric learning and/or for metric application is stored in the RAM 77. This enables more flexibility, in particular when the metric learning and/or application functionalities are not embedded originally in apparatus 7.

Naturally, the present disclosure is not limited to the embodiments previously described.

In particular, the present disclosure extends to any device implementing the described metric learning and/or metric application method. The implementations described herein may take the form of, for example, a method or a process, an apparatus, a software program, a data stream, or a signal. Even if only discussed in the context of a single kind of implementation (for example, discussed only as a method or a device), the implementation of features discussed may also be implemented in other kinds (for example a program).

An apparatus may be implemented in, for example, appropriate hardware, software, and firmware. A relevant apparatus may include a web server, a set-top box, a laptop, a personal computer, a cell phone, a PDA, and other communication devices. As should be clear, the equipment may be mobile and even installed in a mobile vehicle.

The methods may be implemented in an apparatus such as, for example, a processor, which refers to processing devices in general, including, for example, a computer, a microprocessor, an integrated circuit, or a programmable logic device. Additionally, they may be implemented by instructions being performed by a processor, and such instructions (and/or data values produced by an implementation) may be stored on a processor-readable medium such as, for example, an integrated circuit, a software carrier or other storage device such as, for example, a hard disk, a compact disc (“CD”), an optical disc (such as, for example, a DVD, often referred to as a digital versatile / video disc), a RAM or a ROM. Instructions may form an application program tangibly embodied on a processor-readable medium. They may be, for example, in hardware, firmware, software, or a combination.

A processor may be characterized as, for example, both a device configured to carry out a process and a device that includes a processor-readable medium (such as a storage device) having instructions for carrying out a process. Further, a processor-readable medium may store, in addition to or in lieu of instructions, data values produced by an implementation.

As will be evident to one of skill in the art, implementations may produce a variety of signals formatted to carry information that may be, for example, stored or transmitted. The information may include, for example, instructions for performing a method, or data produced by one of the described implementations. Such signals may be formatted, for example, as an electromagnetic wave (for example, using a radio frequency portion of spectrum) or as a baseband signal. The formatting may include, for example, encoding a data stream and modulating a carrier with the encoded data stream. The information that the signals carry may be, for example, analog or digital information. The signals may be transmitted over a variety of different wired or wireless links, and may be stored on a processor-readable medium.

A number of implementations have been described. Nevertheless, it will be understood that various modifications may be made. For example, elements of different implementations may be combined, supplemented, modified, or removed to produce other implementations. Additionally, one of ordinary skill will understand that other structures and processes may be substituted for those disclosed and the resulting implementations will perform at least substantially the same function(s), in at least substantially the same way(s), to achieve at least substantially the same result(s) as the implementations disclosed. Accordingly, these and other implementations are contemplated by this application.

CLAIMS

1. Device (1, 2, 7) adapted for learning a distance metric (*D*) with at least one set of training images representative of at least two objects and with associated pairwise information (y_{st}) indicating whether subsets ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$; $\mathbf{s}$, $\mathbf{t}$) of said training images are pairwise corresponding to same of said objects, characterized in that said device comprises at least one processor configured for:

- associating said distance metric (*D*) with adjustable projection spaces (L_l), adapted to projecting said training images into any of said adjustable projection spaces and to computing local distances ($\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|$; $\|L_l(\mathbf{s} - \mathbf{t})\|$) in said adjustable projection space corresponding to said projected training images,
- obtaining said distance metric (*D*) for said training images as said local distances by selecting, in function of pairs of said subsets of said training images, said adjustable projection spaces corresponding respectively to said distance metric applied to said pairs of said subsets,
- determining adjusted projection spaces from respectively said adjustable projection spaces by reducing offsets between expected values based on said pairwise information (y_{st}) and effective values, of said distance metric (*D*) with respect to said objects for said training images.

2. Device (1, 2, 7) according to claim 1, characterized in that the number (k) of said adjustable projection spaces (L_l) is predefined.

3. Device (1, 2, 7) according to any of claims 1 or 2, characterized in that said at least one set of training images comprising training tubes ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$) of images, said at least one processor is configured for:

- projecting said training tubes into said adjustable projection spaces (L_l),
- computing said local distances ($\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|$) between said projected training tubes,
- obtaining said distance metric (*D*) for each pair of said training tubes by selecting one of said adjustable projection spaces in function of the local distances between the projected training tubes of said pair.

4. Device (1, 2, 7) according to claim 3, characterized in that said at least one processor is configured for computing said local distances between two of said projected training tubes by:

- selecting two projected training images (p, q) of respectively said training tubes in function of local distances $(\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|)$ between the projected training images of said training tubes,
- computing said local distances between said selected projected training images.

5. Device (1, 2, 7) according to any of claims 3 or 4, characterized in that said at least one processor is configured for obtaining said distance metric (D) for any pair of said training tubes ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$) by minimizing said local distances $(\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|)$ between said projected training images belonging to said training tubes over said projection spaces (L_l).

6. Device (1, 2, 7) according to any of claims 1 or 2, characterized in that said at least one processor is configured for:

- computing said local distances $(\|L_l(\mathbf{s} - \mathbf{t})\|)$ between said projected training images,
- obtaining said distance metric (D) for each pair of said training images ($\mathbf{s}$, $\mathbf{t}$) by selecting one of said projection spaces (L_l) in function of the local distances between the respective projected training images of said pair.

7. Device (1, 2, 7) according to any of the preceding claims, characterized in that said at least one processor is configured for determining said adjusted projection spaces (L_l) by proceeding with successive iterations in applying modifications to said adjustable projection spaces, based respectively on successive groups of sample images extracted from said training images.

8. Device (1, 2, 7) according to claim 7, characterized in that said at least one processor is configured for implementing a minimization of a hinge loss ($\mathcal{L}$) based on said groups of sample images, involving said distance metric (D) applied to said groups of sample images and said effective values of said distance metric with respect to said objects for said sample images.

9. Device (1, 2, 7) according to any of the preceding claims, characterized in that said at least one processor is configured for:

- representing said training images by training feature elements ($\mathbf{s}_p, \mathbf{t}_q$; $\mathbf{s}, \mathbf{t}$), each having a determined number of feature dimensions (d),

- determining a basis having a limited number of basis dimensions with respect to said determined number of feature dimensions, founded on said training feature elements,

- projecting said training feature elements onto said basis having said limited number of basis dimensions.

10. Device (1, 2, 7) according to claim 9, characterized in that said at least one processor is configured for initializing each of said projection spaces (L_l) within said basis having said limited number of basis dimensions, by means of a respective limited number of said training images.

11. Device (1, 3) adapted for applying a distance metric (*D*) to at least one set of study images representative of at least two objects, said device being configured for applying pairwise said distance metric to subsets ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$; $\mathbf{s}$, $\mathbf{t}$) of said study images, for comparing resulting values to at least one threshold, and for deciding that said subsets of said study images correspond pairwise to a same of said objects if said resulting values are lower than said threshold,

characterized in that said device comprises at least one processor configured for:

- associating said distance metric (*D*) with projection spaces (L_l), adapted to projecting said study images into any of said projection spaces and to computing local distances ($\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|$; $\|L_l(\mathbf{s} - \mathbf{t})\|$) in said projection space corresponding to said projected study images,

- obtaining said distance metric (*D*) for said study images as said local distances by selecting, in function of pairs of said subsets of said study images, said projection spaces corresponding respectively to said distance metric applied to said pairs of said subsets,

said device (1, 3) adapted for applying a distance metric being preferably configured for exploiting said distance metric obtained by the device (1, 2, 7) adapted for learning a distance metric (*D*) according to any of the preceding claims.

12. Face detection system (1, 2, 3, 7) including at least one device according to any of the preceding claims.

13. Method for learning a distance metric (D) with at least one set of training images representative of at least two objects and with associated pairwise information (y_{st}) indicating whether subsets ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$; $\mathbf{s}$, $\mathbf{t}$) of said training images are pairwise corresponding to same of said objects, characterized in that said method comprises:

- associating said distance metric (D) with adjustable projection spaces (L_l), adapted to projecting said training images into any of said adjustable projection spaces and to computing local distances ($\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|$; $\|L_l(\mathbf{s} - \mathbf{t})\|$) in said adjustable projection space corresponding to said projected training images,
- obtaining said distance metric (D) for said training images as said local distances by selecting, in function of pairs of said subsets of said training images, said adjustable projection spaces corresponding respectively to said distance metric applied to said pairs of said subsets,
- determining adjusted projection spaces from respectively said adjustable projection spaces by reducing offsets between expected values based on said pairwise information (y_{st}) and effective values, of said distance metric (D) with respect to said objects for said training images,

said method for learning a distance metric being preferably implemented by a device (1, 2, 7) for learning a distance metric according to any of claims 1 to 10.

14. Method for applying a distance metric (D) to at least one set of study images representative of at least two objects, said method including applying pairwise said distance metric to subsets ($\bar{\mathbf{s}}$, $\bar{\mathbf{t}}$; $\mathbf{s}$, $\mathbf{t}$) of said study images, comparing resulting values to at least one threshold, and deciding that said subsets of said study images correspond pairwise to a same of said objects if said resulting values are lower than said threshold,

characterized in that said method comprises:

- associating said distance metric (D) with projection spaces (L_l), adapted to projecting said study images into any of said projection spaces and to computing local distances ($\|L_l(\mathbf{s}_p - \mathbf{t}_q)\|$; $\|L_l(\mathbf{s} - \mathbf{t})\|$) in said projection space corresponding to said projected study images,
- obtaining said distance metric (D) for said study images as said local distances by selecting, in function of pairs of said subsets of said study

images, said projection spaces corresponding respectively to said distance metric applied to said pairs of said subsets,
said method for applying a distance metric being preferably implemented by a device (1, 3) for applying a distance metric according to claim 11.

15. Computer program comprising software code adapted to perform steps of a method compliant with any of claims 13 or 14.

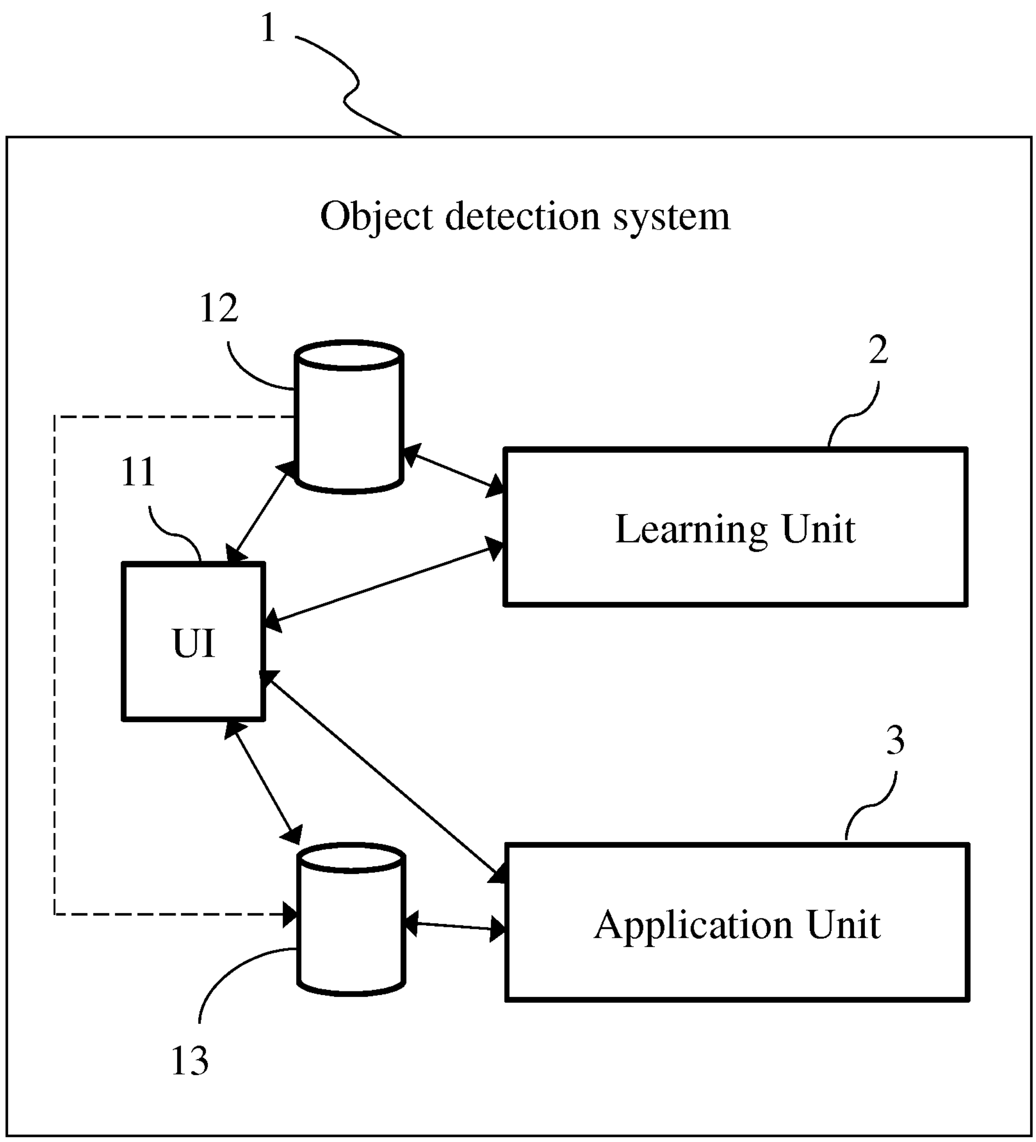
1
Object detection system
12
2
11
Learning Unit
UI
3
Application Unit
13

Fig 1

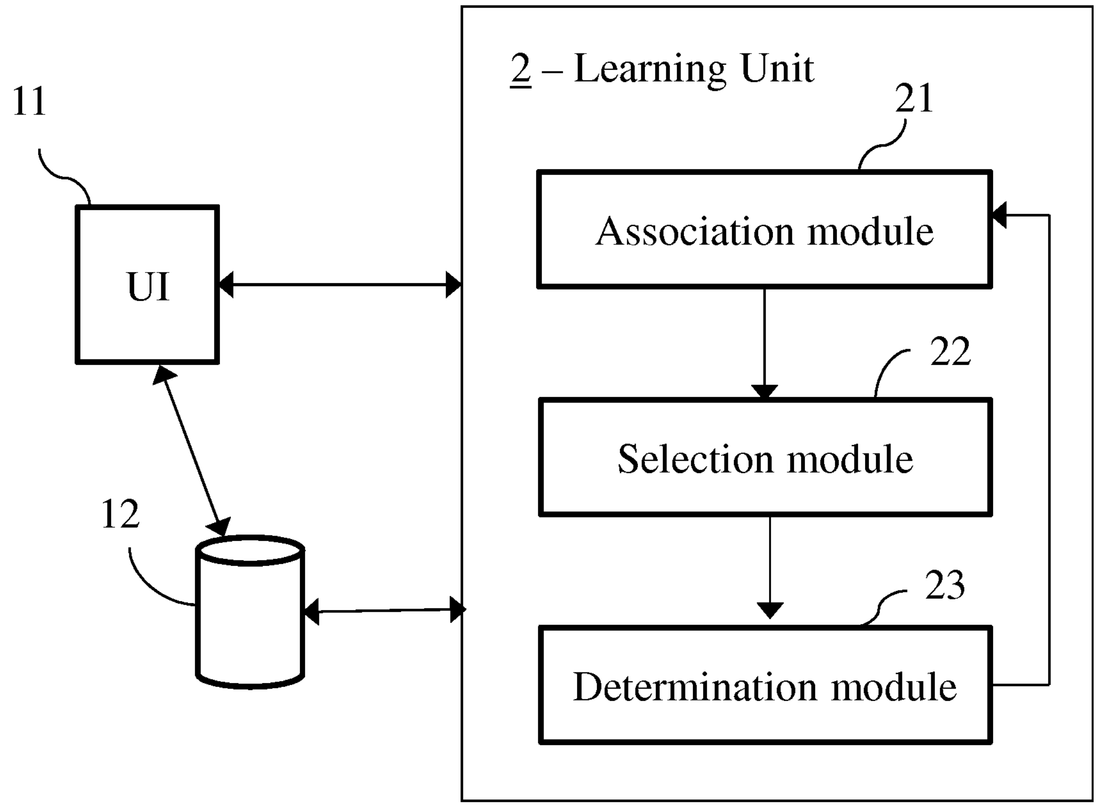
2 – Learning Unit
11
21
UI
Association module
22
Selection module
12
23
Determination module

Fig 2

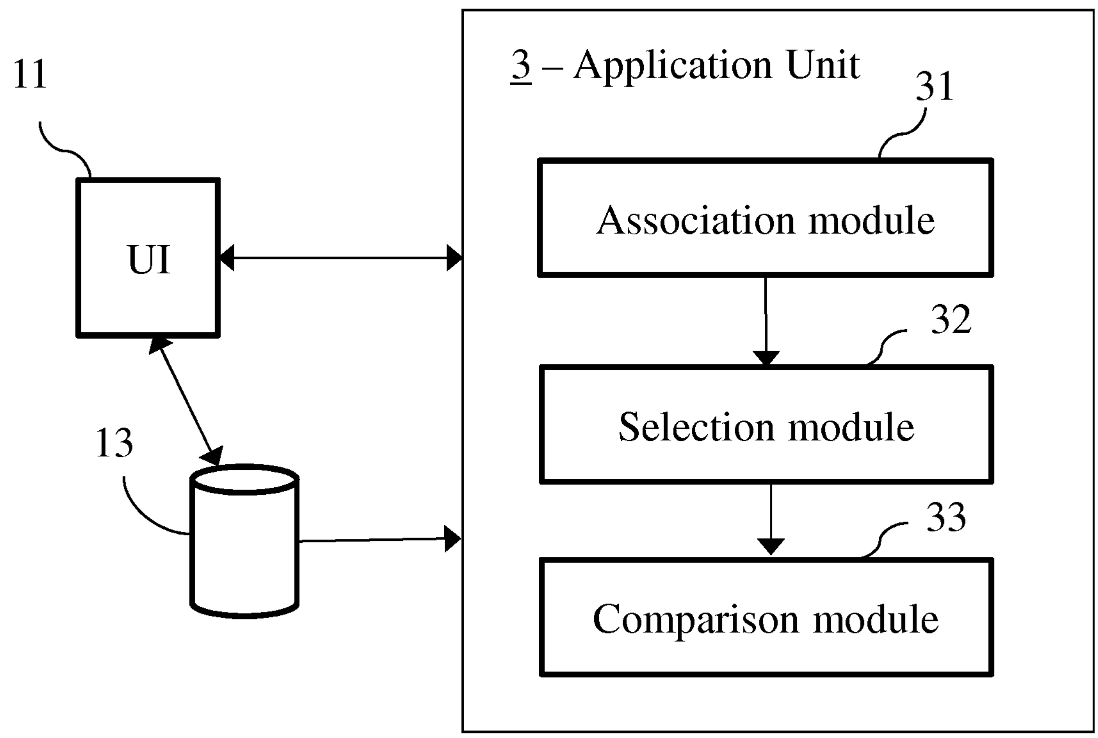
3 – Application Unit
11
31
UI
Association module
32
Selection module
13
33
Comparison module

Fig 3

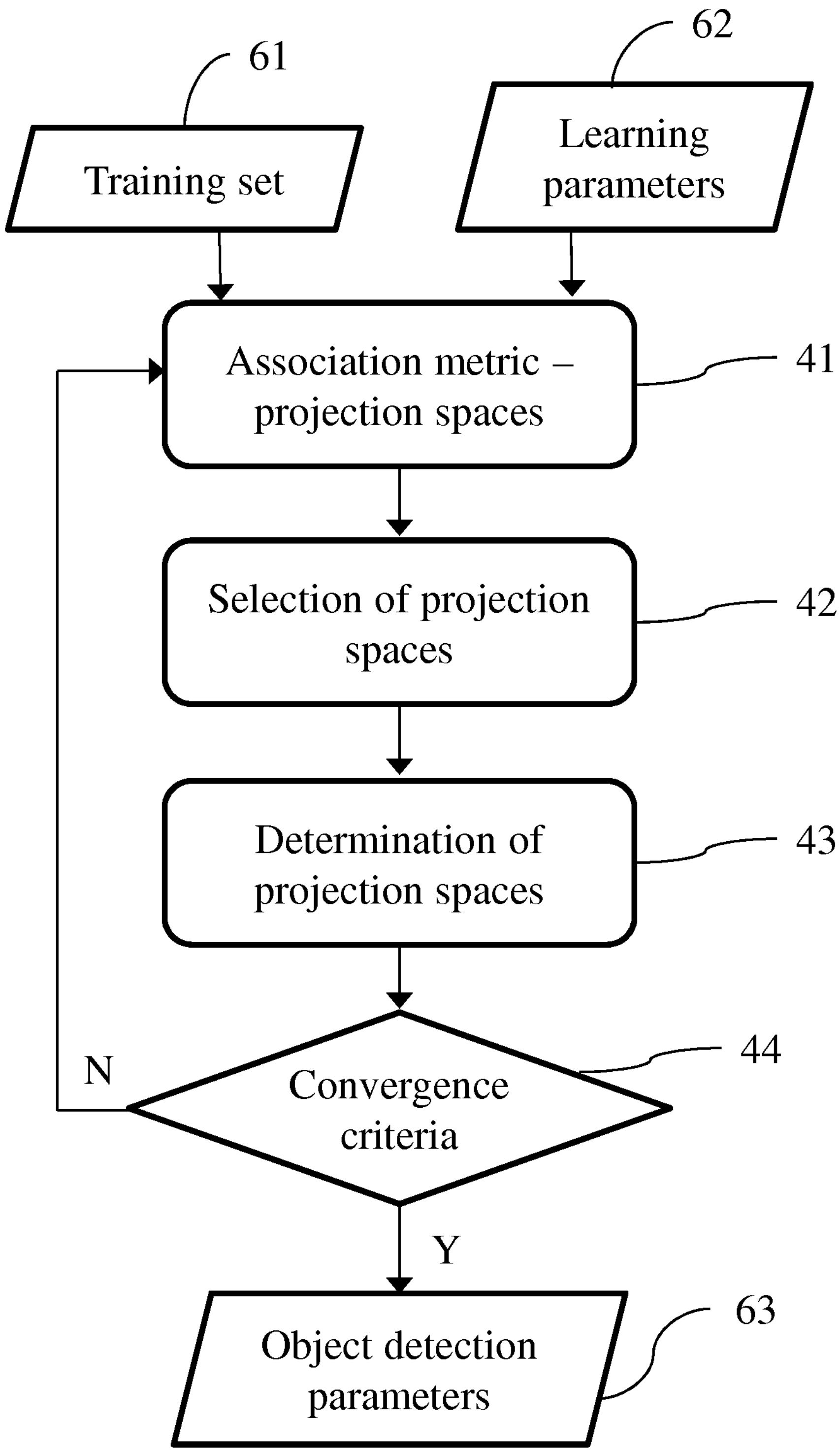
61
Training set
62
Learning parameters
Association metric – projection spaces
41
Selection of projection spaces
42
Determination of projection spaces
43
Convergence criteria
44
N
Y
Object detection parameters
63

Fig 4

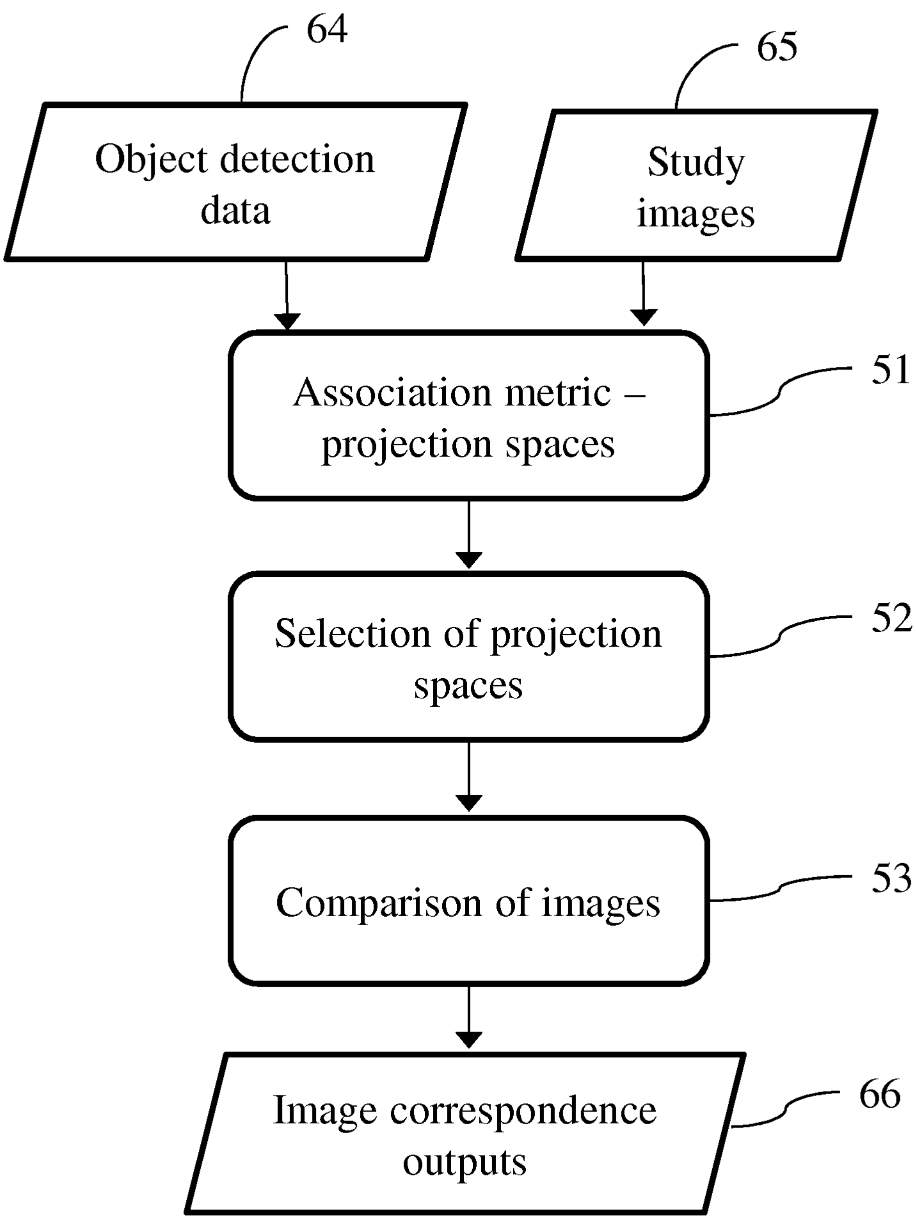
64
Object detection data
65
Study images
Association metric – projection spaces
51
Selection of projection spaces
52
Comparison of images
53
Image correspondence outputs
66

Fig 5

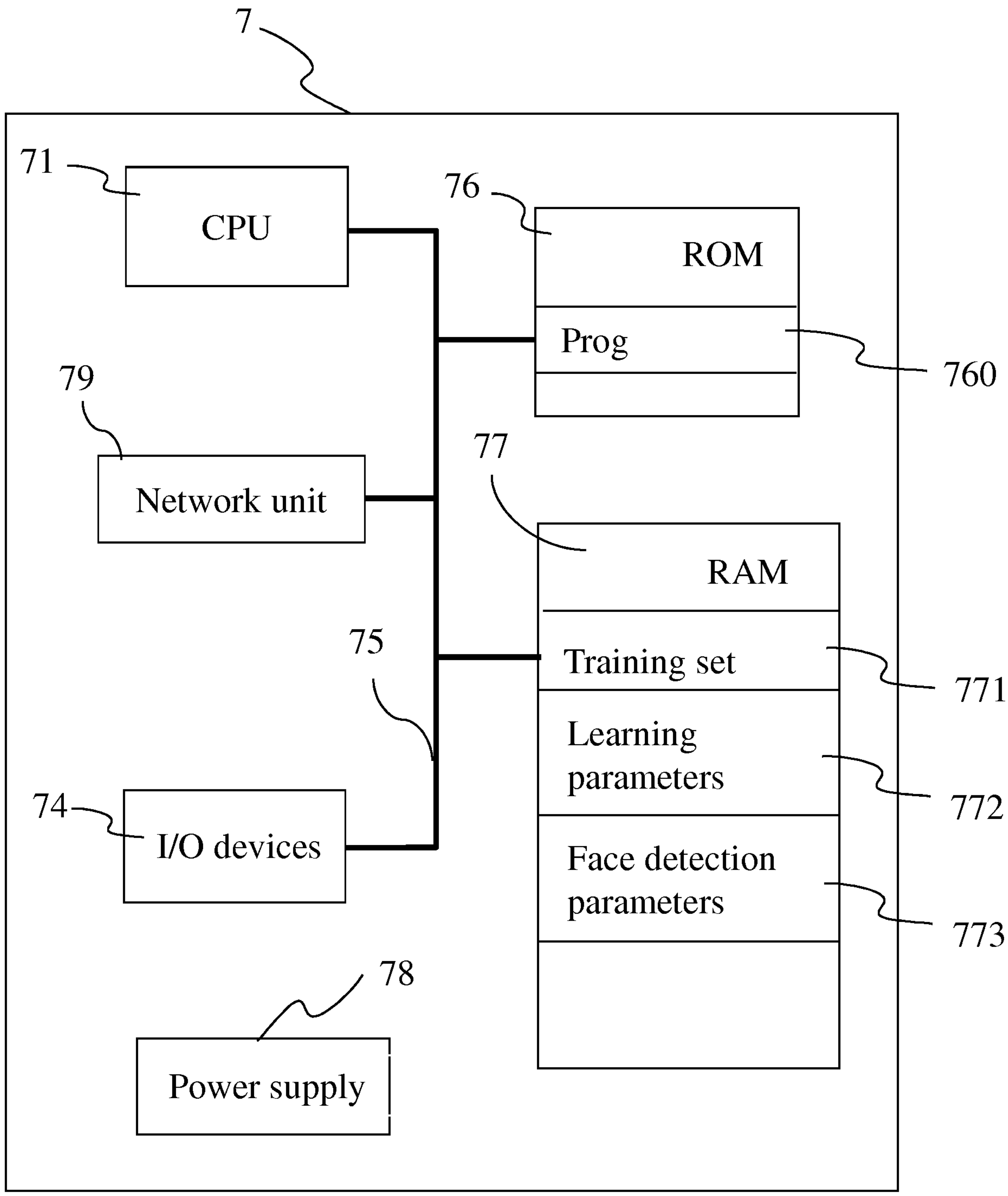
7
71
CPU
76
ROM
Prog
760
79
Network unit
77
RAM
75
Training set
771
Learning parameters
772
74
I/O devices
Face detection parameters
773
78
Power supply

Fig 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2015/051169

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06K9/00 G06K9/62
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06K

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

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

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	Patrick Pérez: "Comparing faces with applications", Inria Vision MicroWorkshop, 2 October 2014 (2014-10-02), pages 1-32, XP055182669, Retrieved from the Internet: URL:http://fire-id.gforge.inria.fr/microworkshop/slides/2014_09_Inria_Vision_microWorkshop_perez_face2face.pdf [retrieved on 2015-04-13] page 14 - page 16 ----- -/--	1-15

[X] Further documents are listed in the continuation of Box C.

[ ] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 April 2015	23/04/2015

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Boltz, Sylvain

1

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2015/051169

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MARTIN KÖSTINGER ET AL: "Synergy-Based Learning of Facial Identity", 28 August 2012 (2012-08-28), PATTERN RECOGNITION, SPRINGER BERLIN HEIDELBERG, BERLIN, HEIDELBERG, PAGE(S) 195 - 204, XP047013787, ISBN: 978-3-642-32716-2 Section 2.1 Section 3.2 -----	1-15
X,P	HUANG LIKUN ET AL: "Multi-manifold metric learning for face recognition based on image sets", JOURNAL OF VISUAL COMMUNICATION AND IMAGE REPRESENTATION, vol. 25, no. 7, 1 September 2014 (2014-09-01), pages 1774-1783, XP029068134, ISSN: 1047-3203, DOI: 10.1016/J.JVCIR.2014.08.006 Section 3 -----	1-15
X	Shijie Xiao ET AL: "Weighted Block-Sparse Low Rank Representation for Face Clustering in Videos" In: "LECTURE NOTES IN COMPUTER SCIENCE", 1 January 2014 (2014-01-01), Springer Berlin Heidelberg, Berlin, Heidelberg, XP055182475, ISSN: 0302-9743 ISBN: 978-3-54-045234-8 vol. 8694, pages 123-138, DOI: 10.1007/978-3-319-10599-4_9, Section 3.2 -----	1-15
X	ZHANG GUANWEN ET AL: "Adaptive Metric Learning in Local Distance Comparison for People Re-identification", 2013 2ND IAPR ASIAN CONFERENCE ON PATTERN RECOGNITION, IEEE, 5 November 2013 (2013-11-05), pages 196-200, XP032581774, DOI: 10.1109/ACPR.2013.86 [retrieved on 2014-03-25] Section II -----	1-15

-/--

1

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Aurélien Bellet ET AL: "A Survey on Metric Learning for Feature Vectors and Structured Data", ArXiv Technical report, 12 February 2014 (2014-02-12), pages 1-59, XP055182418, Retrieved from the Internet: URL:http://arxiv.org/abs/1306.6709 [retrieved on 2015-04-13] Section 3.5 Section 4.3 -----	1-15

1