Algorithms of Two-Dimensional Projection of Digital Images in Eigensubspace: History of Development, Implementation and Application[1]

G. A. Kukharev and N. L. Shchegoleva*

Saint Petersburg Electrotechnical University, St. Petersburg, 197376 Russia

**e-mail: stil_hope@mail.ru*

Abstract—Algorithms for projection of digital images into their eigensubspaces in the framework of linear methods PCA, LDA, PLS and CCA are considered. The history of these methods development of over the past 100 years is given against the backdrop of the emergence of new areas of their application and changing requirements in relation to them. It is shown that this development was initiated by four basic requirements stemming from modern tasks and practice of digital image processing and, first of all, face images (FI). The first requirement is the use of PCA, LDA, PLS and CCA methods in conditions of both a small and extremely large samples of ILs in the initial sets. The second requirement is related to the criterion that determines its eigenbasis, and which should provide, for example, the minimum error of FI approximation, the improvement of clustering in its eigensubspace or the maximum correlation (covariance) between data sets in the subspace. The third one is related to the possibility of applying the methods under consideration to the tasks of processing two or more sets of images from different sensors or several sets of any number matrices. These three requirements led to the emergence, development and application of methods of two-dimensional projection into their eigensubspaces – 2DPCA, 2DLDA, 2DPLS and 2DCCA. Several basic branches of algorithmic implementation of these methods are considered (iterative, not iterative, based on SVD, etc.), their advantages and disadvantages are evaluated, and examples of their use in practice are also shown. Finally, the fourth requirement is the possibility of realizing two-dimensional projections of FI (or other numerical matrices) directly in the layers of convolutional neural networks (CNN/Deep NN) and/or integrating their functions into NN by separate blocks. The requirement and examples of its solution are discussed. Estimates of computational complexity for the presented algorithms and examples of solving specific problems of image processing are given.

Keywords: a set of face images and numeric matrices, an eigenbasis and eigensubspaces, principal components analysis (PCA), linear discriminant analysis (LDA), partial least squares (PLS), canonical correlation analysis (CCA), Karunen-Loev transformation (KLT), 2DPCA/2DKLT, 2DPLS/2DKLT, 2DCCA/2DKLT, CNN, Deep NN

DOI: 10.1134/S1054661818020116

INTRODUCTION

The Origins of Projection Methods into the Eigensubspace

The main ideas of methods of projections onto the eigensubspaces, as one of the tools for mathematical processing of observations and the tool for identifying and/or establishing links in observations, were founded in [1–4]. Approximation problem solution of experimental data presented on the plane by a set of points was shown in [1], using the original criterion, in which the approximation error minimum is always achieved. Relying on this result, a method for finding the **principal components** in a set of variables was presented in [2], where each variable was represented by a separate vector of empirical data. And, it is here that the concept of eigenvectors and eigenvalues, the Component Analysis (PCA) algorithm and the projection of the original data into a proper subspace are introduced. In [3], the problem of linear discriminant analysis is discussed – finding a function that determines the best separation of data populations in an eigensubspace. To solve this problem, was proposed a criterion that minimizes the intra-class and maximizes the interclass distance in the subspace, which can be considered as an improvement in clustering in it. An algorithm for constructing an eigenbasis, based on this criterion, the choice of the principal components, and the procedure of projection into an eigensubspace was described. Solutions [3] created the possibility of more efficient solution of image recognition problems with a simplified structure of classifiers, since the division of data in the subspace was performed best of all possible ways. The criterion for maximizing cross-cor-

[1] The article is published in the original.

Received November 10, 2017

relation in the eigensubspace for two independent (in the general case) data sets and the necessary algorithm for finding an eigenbasis common to two sets of input data was presented in [4]. Solutions [4] created the possibility of detailed development of Canonical Correlation Analysis (CCA) and Partial Least Squares (PLS), as methods of projection into a common eigenspace for two or more sets of input data.

All methods of projection into the eigenspace mentioned above are realized in two consistent stages. The first stage involves eigen basis building for the source data and selecting the principal components. The greatest computational costs are associated with the solution of eigenvalue problems. And, if the dimension of the original feature space is D, then the covariance matrices and common scattering matrices (in accordance with the required criterion) whose order is D are calculated first, and then D eigenvalues and D corresponding eigenvectors are calculated, and are also determined d principal components ($d \ll D$). In general, the computational cost here is $O_1 \cong D^3$.

At the second stage, the projection of the original data into the eigenspace (as a transformation in the intrinsic basis or the Karunen-Loeve transformation) is realized and the dimension space of the characteristics is reduced to the values of d. At the same time, computational costs here are $O_2 \leq D^2$.

1. PROBLEMS OF PCA IMPLEMENTATION FOR IMAGES

In the problems of building their eigenbases for digital image sets, five basic parameters are taken into account: $\{M, N, K, L, Q\}$. Here: M and N are the number of rows and columns of the source images; K and L determine the number of image classes in the source data set and the number of images in each class respectively; Q is the number of sets of images. In general, $L \geq 1$ and $Q \geq 1$, and if $L = 1$, then such data are defined as unstructured, and if $Q = 1$, then all source images are placed in one set of input data.

The parameters M and N specify the dimension D of the original feature space so that $D = MN$ and determine the computational complexity of the algorithms being implemented. In this case, the problem of realizing the projection of images into their eigensubspaces is accompanied by two problems. The first of them is the problem of the small number of images in a set in comparison with the dimension D, which is defined as a **small sample problem** when $D \gg KL$. The second is the problem of an **extremely large sample**, that is, when $KL \gg D \gg 1000$ (for example, KL can range from several thousand to a million images, which is typical for modern face databases or multimedia databases).

Recall that in the approaches [2–4], all these operations were performed for sets of input data presented in vector form. Therefore, the first application, for example, of PCA and LDA methods for digital images, appeared almost at the end of the 20th century [5–7], and the methods of CCA and PLS only in the 21st century, which was associated with the impossibility of direct transfer of the [2–4] on images (considered as two-dimensional data structures), and the large amount of computations that arise in this case.

Considering this, the methodology of using PCA in an application to images processing based on need to use basic PCA method [2], originally oriented to processing of vector data, and on the other hand, with the fulfillment of the condition

$$D < KL, \tag{1}$$

to ensure the stability of solving eigenvalue problems. In this case, each original image was transformed by concatenating its columns (or rows) into a common vector of size $MN \times 1$, and then the entire set of input data already contained KL of such vectors. Further, several approaches were applied to the original data set, “allowing to maneuver” between the above constraints and condition (1).

If, for example, a small sample problem occurs (i.e., condition (1) is not satisfied), and the value $KL \ll 1000$, then the entire set $\{X\}$ of the source images is represented by a matrix of size $MN \times KL$. In this case, instead of usual covariance matrices, the Gram matrices G is used, of order KL so that $G = X^{\mathrm{T}}X$. Further, for the matrix G, the eigenvalue problem was solved, which ensured the computation of the KL eigenvalues and the corresponding eigenvectors (each with the size $KL \times 1$), which could be recalculated into the whole composition of the eigenvectors [5].

Wherein, the amount of computation was significantly reduced. The computational costs at the stage of solving the eigenvalue problem in this case are $O \cong (KL)^3$. The projection into the eigensubspace is realized as an one-dimensional Karunen-Loeve transformation (1D KLT) of the vectorized initial data by multiplying them by the projection matrix with cost $O \cong (KL)^2$.

If condition (1) was fulfilled, but the dimension D of the original feature space was large (for example, $D \gg 1000$), the solution was based on decreasing each original image to size $m < M$ and $n < N$, and then converting it to a common vector of size $mn \times 1$. The set of such vectors (centered relative to the set average vector) formed a covariance matrix of order mn. The projection matrix (also of order of mn) in the subspace was further defined on the basis of the solution of the eigenvalue problem for the covariance matrix. The rows of the projection matrix (size $mn \times 1$) are the desired eigenvectors of the covariance matrix. In this case, the projection into the eigensubspace is realized as a one-dimensional Karunen-Loeve transformation (1D KLT) of vectors (representing the centered initial data) by multiplying them by the projection matrix.

The values of m and n must be chosen from the stability condition for the solution of the eigenvalue problem. Algorithmically, the task of reducing the original feature space of $MN \to mn$ for source images is solved using the "resize", "down sampling" procedures or by moving from the source space of luminance features to another space – for example, in a brightness histogram, luminance gradients [6], spectrum discrete Fourier transform [3], etc. The computational costs of the eigenvalue problems in these cases is $O \cong (mn)^3$, but the one-dimensional Karunen-Loeve transformation is $O \cong (mn)^2$.

It should be remembered that reducing the resolution of images associated with its physical size to $m \times n$ can lead to loss of information in them. Therefore, if the problem of an extremely large sample arises, and the previous solution with a decrease in the original feature space is not permissible due to the loss of important information in the original images, then it is impossible to achieve the practical solution of the problem of constructing an eigenbasis for digital images within the framework of ideas [2]! Finally, it should be noted that in the "artificial vectorization" of images, the spatial relationships of neighboring elements do not persist in vectors. The possibility of full use of the correlation found in the original matrices between neighboring elements is lost too.

Naturally, these problems – the inability to use PCA for images at the boundary values of the basic parameters, the loss of information and the loss possibility of the full use of the correlation between the source elements – also apply to the LDA, PLS and CCA methods.

However, the problem solution unexpectedly appeared in 1998 with a new approach to the implementation of the Karunen-Loeva transformation in the application to the face images. The approach was named as "Vector Based Approximation of KLT" – VKLT [8]. Its main idea is that unlike [2, 5, 6] in VKLT two covariance matrices C_0 and R_0 are calculated, which are determined separately by the columns and rows of all the original images without their concatenation into a vector. Next, we compute the two projection matrices $\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2$ defined by the eigenvectors of the covariance matrices C_0 and R_0. The projection of the original images into new feature space is realized as a two-dimensional (in rows and columns) Karunen-Loeve transformation so that

$$Y_k = \Phi_1^T X_k \Phi_2 \ , \quad \text{for} \quad \forall \quad k = 1, 2, \dots K, \tag{2}$$

where X_k and Y_k – image and its projection in its eigenspace, which distinguishes approach [8] from [2, 5, 6]. We note that in (2) the original image X_k and its projection Y_k in the subspace are represented by matrices of size $M \times N$.

The experiments on the face recognition described in [8] were performed on the ORL database [24]. At the same time, the authors noted the highest robustness of the VKLT method (in comparison with KLT) in relation to the quality of the input images, including their "noisiness", cyclic shift and brightness changes. Also, there was a significant decrease in the required amount of memory and computational costs when implementing eigenvalue problems (i.e., when computing projection matrices $\mathbf{\Phi}_1$ and $\mathbf{\Phi}_2$).

The ideas of [8] are presented in monographs [9–11] in the form of PCArc (PCA on rows and columns of images) and investigated in the application for faces recognition *in small sample size* (SSS), as well *as in low resolution* and additive noise of test images. In [9] it was shown for the first time that for any $K \geq 1$ the SSS problem does not arise in the PCArc framework, and the robust properties noted in [8] are preserved. Also, examples of displaying face images of the principal components in 3D subspaces for PCArc and (PCArc + LDA) were considered. In [10], "Face Recognition System Modeler/FaRes-MOD" package was also presented, in which it is *no iterative algorithms of the two-dimensional projection* into eigen PCArc and LDArc that are implemented. FaRes-MOD package is implemented on the Borland® C ++ Builder ™ platform, runs under Windows OS and is ported to all its modern versions. Since 2003, the package has been used at the Polytechnic University of Szczecin in Poland and the St. Petersburg State Electrotechnical University (LETI) for training purposes in the courses "Computer methods of person identification". Vector-matrix procedures are presented in [11] that allow one to directly implement methods of two-dimensional projection and reduction of the dimension of the feature space in algorithmic languages supporting matrix operations. Methods for improving clustering in our eigensubspace by using the cascade of PCArc + LDA projection methods are discussed.

Further development of ideas [8–11] was presented in publications as methods 2DLDA/2DKLT ([12] – May 2005, and also [13–15]) and 2DPCA/2DKLT [14, 16]. In this form of recording, for the first time in the technical literature, two stages of the implementation of these methods are emphasized: the first stage is an analysis of the initial data (including the formation of common scattering matrices by a given criterion, the calculation of the projection matrices and the determination of the principal components) and the second stage following it, the two-dimensional Karunen- Loeva. In addition, this form of recording distinguishes the methods presented in [8–16] from other 2DLDA and 2DPCA methods implemented by iterative algorithms. In addition, this form of recording distinguishes the methods presented in [8–16] from other 2DLDA and 2DPCA methods implemented by iterative algorithms. And so, in publications [12–16] a comparative analysis of the methods of 1D and 2D projection into eigensubspaces is presented and examples of solving problems of clustering, compression and recognition of images of individuals on various benchmark databases are shown. In [14] programs for implementing these methods in the language of the MATLAB® package are presented. In [15], a system for recognizing faces is shown using the 2DLDA/2DKLT method if there is **only one (!) image** in each class of the training base. The presented solution in practice demonstrates the possibility of “bypassing the SSS problem”, usually accompanying the tasks of processing images of individuals. The non-iterative algorithms of the parallel and cascade form of implementing the 2DPCA/2DKLT and 2DLDA/2DKLT methods are described in [16]. Finally, in [16] all the characteristics of the 2DPCA/2DKLT method were also presented for the first time, which will be presented after its formal description.

2. A FORMAL DESCRIPTION OF THE 2DPCA/2DKLT METHOD FROM [8–16]

Let X be a set consisting of K matrices, where each matrix represents an image of size $M \times N$, where $MN \gg K$:

$$X = [X^{(1)}X^{(2)}\ldots X^{(K)}], \quad \forall k = 1,2,\ldots K. \tag{3}$$

The purpose of the 2DPCA is to define two projection matrices W_1 and W_2 that transform the input data (3) into eigensubspace of features when the minimum error of approximation condition is met, which can be written as follows:

$$\left\| X^{(k)} - \hat{X}^{(k)} \right\| \to \min, \quad \forall k, \tag{4}$$

where $\hat{X}^{(k)}$ – the approximation result of the k-th image by principal components. Versions of the parallel and cascade algorithms 2DPCA/2DKLT are shown in Table 1.

The feature space dimensionality reduction is realized as a “truncated 2DKLT”, in which only eigenvectors that correspond to the d principal components participate in further transformations. For this, from the matrix W_1 select d rows, and from the matrix W_2 select d columns corresponding to d largest eigenvalues, and on their basis form two reduction matrices F_2 and F_2. Wherein $d < \min(M, N)$ or $d_1 < M$; $d_2 < N$, if $d_1 \neq d_2$.

The upper bounds of the parameter d are determined based, for example, on the criterion of energy significance of the eigenvalues, and the lower bounds are chosen taking into account the desired quality of the approximation of the original data. In this case, the “truncated two-dimensional Karunen-Loeve transformation” is realized as follows:

$$\hat{Y}^{(k)} = F_1 \bar{X}^{(k)} F_2, \quad \forall k. \tag{5}$$

The matrices F_1 and F_2 in (5) have dimensions $(d \times M)$ and $(N \times d)$, respectively, or $(d_1 \times M)$ and $(N \times d_2)$ in the general case. The sign “^” – determines the difference between the result of the approximation and the result of the reconstruction $Y^{(k)}$. In this case, the result (5) is a matrix of size $d_1 \times d_2$, represents the original images in the eigensubspace of the features. Now condition (4) can be represented in a new form:

$$\left\| X^{(k)} - F_1^{\mathrm{T}} \hat{Y}^{(k)} F_2^{\mathrm{T}} \right\| \to \min, \quad \forall k = 1, 2\ldots K. \tag{6}$$

Figure 1 schematically shows the basic steps of 2DPCA implementation for centered (relative to average) input data: the calculation of two covariance matrices, the solution of the two eigenvalue problems, the selection of two boundaries for the principal components, and the formation of the left F_1 and right F_2 projection matrices.

Figure 2 shows the 2DKLT implementation procedure in a parallel algorithm and the organization of calculations in the cascaded algorithm 2DPCA/2DKLT.

Characteristics of the 2DPCA/2DKLT Method (First Introduced in [14])

Dimensions of the feature space in solving the eigenvalue problems in the 2DPCA/2DKLT method are determined by the number of rows and columns of the images (that is, the M and N values). Therefore, the largest dimension of the original feature space is defined as $\mathbf{max}(M, N)$.

If K source images would be represented as a collection of rows and columns, the total number of vectors obtained is $K(M + N)$. Therefore, for any values of $\{M, N, K\}$, the relation “the dimension of the original feature space/number of vectors” will always correspond to the condition $\mathbf{max}(M, N) \ll K(M + N)$, thereby “bypassing” the SSS problem.

The order of the covariance matrices is M and N, which predetermines the practical possibility of solving the eigenvalue problem and the stability of this solution even for images of relatively large sizes. For covariance matrices computed separately by rows and columns of the same source image, $2MN(M + N)$ operations are required, and for all K images – $2KMN(M + N)$ operations.

The calculations volume for solving eigenvalue problems for an image database with parameters $\{M, N, K, L\}$ is at most $((M)^3 + (N)^3)$ when all eigenvalues and corresponding eigenvectors are found. Note that the operational complexity for the 1DPCA method is determined by the values $(MN)^3$ or $KL)^3$. If, for example, assume that $MN = KL$ and $M = N$, then the calculation reduction for the 2DPCA method in comparison with the 1DPCA method is determined by the ratio $M^3/2$.

The implementation of one complete 2DKLT will require $MN\ (M + N)$ operations, and the $Md(d + N)$ operations (d is the number of principal components) will be required to implement the procedure for the reduction of the feature space dimension (RFSD)

Table 1. Parallel and cascade algorithms for 2DPCA/2DKLT

Parallel algorithms of 2DPCA/2DKLT	Cascade algorithms of 2DPCA/2DKLT
Input: matrices $X^{(k)} \in \Re^{M\times N}, \ \forall k = 1,2,\dots K$.	**Input:** matrices $X^{(k)} \in \Re^{M\times N}, \ \forall k = 1,2,\dots K$.
Output: matrices $\bar{X}$, W_1, W_2, Λ_1, Λ_2, $Y^{(k)}$.	**Output:** matrices $\bar{X}$, W_1, W_2, Λ_1, Λ_2, $Y^{(k)}$.
1. Calculate the average image from all data $\bar{X} = \frac{1}{K}\sum_{k=1}^{K} X^{(k)}$. 2. Center the input data: $\bar{X}^{(k)} = X^{(k)} - \bar{X}, \forall k$ 3. Calculate the two covariance matrices, relative to the rows and columns of the result matrices obtained in p.2: $C^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{X}^{(k)}(\bar{X}^{(k)})^T$, $C^{(\mathrm{c})} = \sum_{k=1}^{K} (\bar{X}^{(k)})^T \bar{X}^{(k)}$. 4. Solve two eigenvalue problems: $C^{(\mathrm{r})}W_1 = \Lambda_1 W_1$ and $C^{(\mathrm{c})}W_2 = \Lambda_2 W_2$ defining the diagonal eigenvalues matrices (Λ_1, Λ_2) and the eigenvectors matrices W_1 and W_2 (or the left and right projection matrices). 5. Arrange the eigenvalues in descending order and rearrange, in accordance with the new order, the columns of the matrices W_1 and W_2. 6. To project the input data into eigenspace, realizing the two-dimensional Karunen-Loeve transformation: $Y^{(k)} = W_1^{\mathrm{T}} X^{(k)} W_2, \ \forall k$ 7. To reconstruct the original data, by the result $Y^{(k)}$, perform the inverse two-dimensional Karunen-Loeve transformation: $X^{(k)} = W_1 Y^{(k)} W_2^{\mathrm{T}}, \ \forall k$. ***The end*** The parameters of all matrices mentioned above $\bar{X} \in \mathrm{R}^{M\times N}; Y^{(k)} \in \mathrm{R}^{M\times N}$; $C^{(\mathrm{r})} \in \mathrm{R}^{M\times M}; \quad \mathrm{C}^{(\mathrm{c})} \in \mathrm{R}^{N\times N}$; $\{W_1, \Lambda_1\} \in \mathrm{R}^{M\times M}; \quad \{W_2, \Lambda_2\} \in \mathrm{R}^{N\times N}$. Matrices Λ_1, Λ_2 are diagonal; Matrices W_1, W_2 – orthogonal in such a way that: $W_1^{\mathrm{T}}W_1 = I_1$; $W_2^{\mathrm{T}}W_2 = I_2$, here I_1, I_2– identity matrices of order M and N. P.S. The matrices calculated in the cascade algorithm also correspond to these parameters.	1. Calculate the average image from all data $\bar{X} = \frac{1}{K}\sum_{k=1}^{K} X^{(k)}$. 2. Center the input data: $\bar{X}^{(k)} = X^{(k)} - \bar{X}, \forall k$ 3. Calculate the covariance matrix with respect to the rows of the result matrices from p. 2: $C^{(\mathrm{r})} = \frac{1}{MN}\sum_{k=1}^{K} \bar{X}^{(k)}(\bar{X}^{(k)})^T$. 4. Solve the first eigenvalue problem: $C^{(\mathrm{r})}W_1 = \Lambda_1 W_1$ defining the diagonal eigenvalues matrix Λ_1 and the eigenvectors matrix W_1 (or the left projection matrix). 5. Arrange the eigenvalues in descending order and rearrange, in accordance with the new order, the columns of the matrix W_1. 6. To project the input data into an intermediate proper subspace, realizing the Karunen-Loeve transformation: $X_1^{(k)} = W_1^{\mathrm{T}} X^{(k)}, \ \forall k$. 7. Calculate the covariance matrix over the columns of the result matrix (6): $C^{(\mathrm{c})} = \sum_{k=1}^{K} (X_1^{(k)})^T X_1^{(k)}$. 8. Solve the second eigenvalue problem: $C^{(\mathrm{c})}W_2 = \Lambda_2 W_2$ and, thus, calculate eigenvalues matrix Λ_2 and the eigenvectors matrix – the right matrix of the projection W_2. 9. Arrange the eigenvalues in descending order and rearrange the columns of the matrix W_2 in accordance with the new order. 10. Perform the projection of the input data into a eigensubspace by implementing Karunen-Loeve transformation in one-dimensional or two-dimensional form: $Y^{(k)} = X_1^{(k)} W_2 \equiv W_1^{\mathrm{T}} X^{(k)} W_2, \ \forall k$ ***The end***

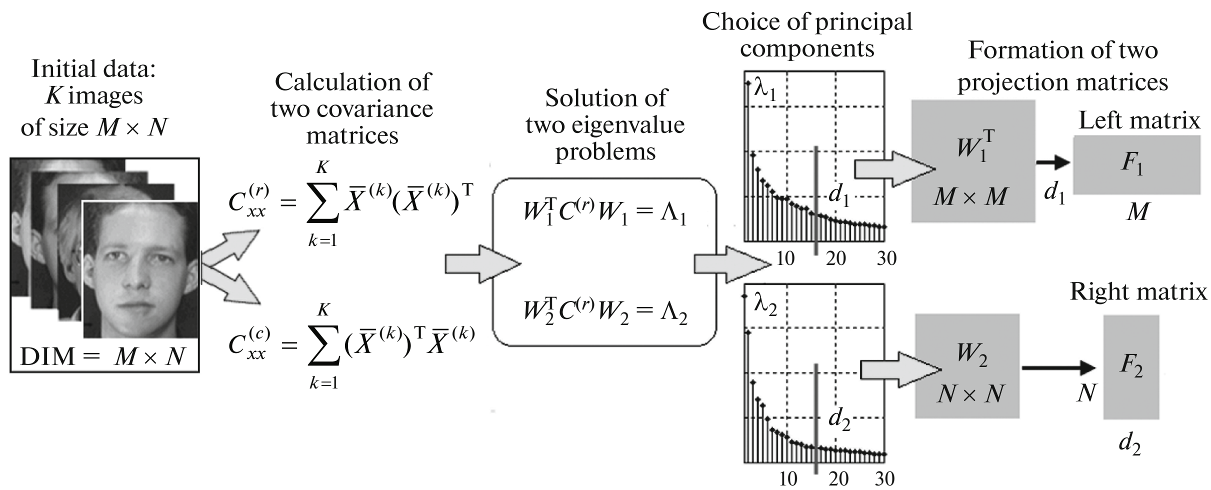


Fig. 1. The main stages of 2DPCA implementation.

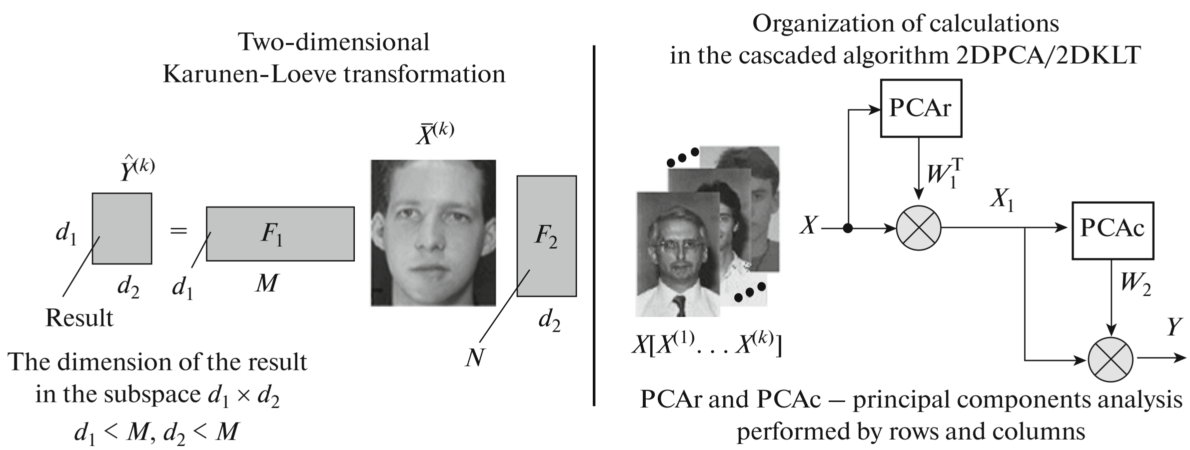


Fig. 2. Variants of calculations organization of 2DPCA/2DKLT method.

within the "truncated" 2DKLT. At the same time, the calculation can be roughly estimated as the value $(M + N)/d$, because:

$$\frac{NM^2 + MN^2}{Md^2 + MNd} = \frac{MN(M+N)}{M(d+N)d} \approx \frac{M+N}{d},$$
$$\text{on condition} \quad d \ll N.$$

For example, for $M = 112$ and $N = 92$ (ORL base [24]) and $d = 10$, the computation reduction will be approximately 20 times (!) for each image. Taking into account the parameter K – the number of images, the acceleration of calculations will amount to approximately $K(M + N)/d$ for all the initial data. The result (RFSD) contains $(d \times d)$ or $(d_1 \times d_2)$ elements, so the RFSD is defined by the ratio $MN/(dd)$ or $MN/(d_1d_2)$, if $d_1 \neq d_2$. For example, for $M = 112$, $N = 92$ and $d = 10$, the feature space will shrink more than 100 times (!) for each image.

3. 2DLDA/2DKLT ALGORITHM IMPLEMENTATION FOR A SET OF IMAGES

Let's consider separately the 2DLDA/2DKLT implementation algorithm for a set of images, first introduced in May 2005 [12]. In this case, the source data must be structured (divided into classes) and in each class there must be at least two images.

Let us specify a set X consisting of KL matrices $X^{(k,l)}$, where each matrix represents an image of size $M \times N$, and $k = 1, 2, \ldots K$ and $l = 1, 2, \ldots L$.

The goal of 2DLDA is to define the left and right projection matrices W_1 and W_2 that transform the original data into a eigensubspace of characteristics such that $X^{(k,l)} \to Y^{(k,l)}$, $\forall k$ and $\forall l$, when the minimization condition of the class is minimized and the intersectional distance in the subspace is maximized. The pseudo-code for the 2DLDA/2DKLT algorithm are shown in Table 2.

Table 2. Pseudo-code for the 2DLDA/2DKLT algorithm

Input: matrices $X^{(k,\, l)} \in \Re^{M \times N}$,
$\forall k = 1, 2, \ldots K$; $\forall l = 1, 2, \ldots L$.

Output: matrices $\bar{X}$, W_1, W_2, Λ_1, Λ_2, $Y^{(k,\, l)}$.

1. Calculate the average image in each class

$$\bar{X}^{(k)} = \frac{1}{L}\sum_{l=1}^{L} X^{(k,\, l)}, \quad \forall k.$$

2. Calculate the average image for all data:

$$\bar{X} = \frac{1}{K}\sum_{k=1}^{K} X^{(k)}.$$

3. Compute the "Within-class" and "Between-class" covariance matrices defined relative to the rows:

$$W^{(\mathrm{r})} = \sum_{k=1}^{K}\sum_{l=1}^{L}(X^{(k,\, l)} - \bar{X}^{(k)})(X^{(k,\, l)} - \bar{X}^{(k)})^T;$$

$$B^{(\mathrm{c})} = \sum_{k=1}^{K}(\bar{X}^{(k)} - \bar{X})^T(X^{(k)} - \bar{X}).$$

4. Compute the "Within-class" and "Between-class" covariance matrices defined relative to the columns:

$$W^{(\mathrm{c})} = \sum_{k=1}^{K}\sum_{l=1}^{L}(X^{(k,\, l)} - \bar{X}^{(k)})^{\mathrm{T}}(X^{(k,\, l)} - \bar{X}^{(k)});$$

$$B^{(\mathrm{c})} = \sum_{k=1}^{K}(\bar{X}^{(k)} - \bar{X})^{\mathrm{T}}(X^{(k)} - \bar{X}).$$

5. Perform regularization of matrices $W^{(\mathrm{r})}$ and $W^{(\mathrm{c})}$ before their inversion and regularization of both scattering matrices:

$S^{(\mathrm{r})} = [W^{(\mathrm{r})}]^{-1}B^{(\mathrm{r})}$ and $S^{(\mathrm{c})} = [W^{(\mathrm{c})}]^{-1}B^{(\mathrm{c})}$.

6. Solve two eigenvalue problems:

$$S^{(\mathrm{r})}W_1 = \Lambda^{(\mathrm{r})}W_1;$$

$$S^{(\mathrm{c})}W_2 = \Lambda^{(\mathrm{c})}W_2.$$

7. Perform projection of the original data into eigenspace, realizing the two-dimensional Karunen-Loeve transformation:

$$Y^{(k,\, l)} = W_1^{\mathrm{T}}\bar{X}^{(k,\, l)}W_2, \quad \forall k$$

To reconstruct the original data, by the result, perform the inverse two-dimensional transformation of Karunen-Loeve:

$$\bar{X}^{(k,\, l)} = W_1 Y^{(k,\, l)} W_2^{\mathrm{T}}, \quad \forall k.$$

The end

Feature space dimensionality reduction

From the matrix W_1^T select d rows, and from the matrix W_2 choose d columns corresponding to d largest eigenvalues, and on their basis form two reduction matrices F_1 and F_2. In this case, the "truncated two-dimensional Karunen-Loeve transformation" is realized as follows:

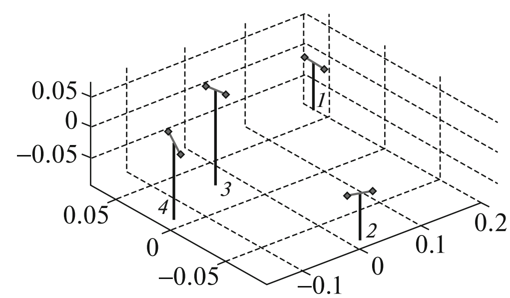
$$\hat{Y}^{(k,\, l)} = F_1\bar{X}^{(k,\, l)}F_2, \quad \forall k,$$

which is similar to the transformations in the 2DPCA/2DKLT algorithm.
The matrices F_1 and F_2 have dimensions ($d \times M$) and ($N \times d$), respectively, or ($d_1 \times M$) and ($N \times d_2$) in the general case. The sign "^" - distinguishes the result of the approximation from the result of the $Y^{(k,\, l)}$ reconstruction.

\- - - - - - - - - - - - -

In general, the original data can represent K sets of images, where each set consists of L_k images so that $L_k \geq 2$, a $K \geq 3$. These two conditions make it possible to calculate the average image in the class, and also to determine at least two characteristics for mapping the proper subspace on the plane. In the conditions of image processing, it is sufficient to have only two images per class. For $K = 4$ and $L = 2$, the proper subspace is shown below:

4.784 0.21 0.02:



The maximum distance in classes here is 0.02, the minimum distance between classes was 0.21, so the clustering quality parameter was 4.78.

The parameters of all matrices calculated above

$$\{\bar{X},\ \bar{X}^{(k)},\ Y^{(k,\, l)}\} \in \Re^{M \times N}; \quad \{C^{(\mathrm{r})}, W_1, \Lambda_1\} \in \Re^{M \times M}; \quad \{C^{(\mathrm{c})}, W_2, \Lambda_2\} \in \Re^{N \times N}.$$

The matrices Λ_1, Λ_2 – are diagonal. The matrices W_1, W_2 – are close to orthogonal.

4. THE DEVELOPMENT OF METHODS FOR THE TWO-DIMENSIONAL PROJECTION OF FACE IMAGES INTO EIGENSUBSPACES

Continuing the history of projection methods into our eigensubspaces, note that some "new approaches to the implementation of 2DPCA" were considered in [17], published after [8–11]. However, in work [17], there are fundamental errors in the presentation of the basic procedure of the 2DPCA. First, in [17], the analysis of the principal components was carried out in **only one direction** – by the columns of the original images (!). Secondly, in a different direction (in rows), analysis of the principal components was not performed, but the size of the image was changed. Naturally, both KLT is realized here only in one direction, as well as the reduction in the dimension of the feature space. Third, to allow the practical application of the approach [17] for face recognition problems, a preliminary reduction in the size of the original images was applied (which is not required in the methods [8–16]). That is why the approaches presented in [17] do not relate to methods of two-dimensional projection into eigensubspaces and cannot be called 2DPCA.

Developing the ideas of the work [17], David Zhang published an article [18] in December 2005, in which he presented a ***mix*** of the concepts "*Two-Directional* and *Two-Dimension*", which, however, further complicated the understanding of the ideas [8–11] projections into their eigensubspaces in an application to image processing.

Nevertheless, the approaches [17] had other followers who developed them further. Their essence boiled down to the preliminary decomposition of the original image into a set of subdomains. Nevertheless, the approaches [17] had other followers who developed them further. Their essence boiled down to the preliminary decomposition of the original image into a set of subregions. On the one hand, this allowed "straightforward" implementation of the "2DPCA [17]" method for such subregions, and, on the other hand, increased the representativeness of the original data, which led to an increase in recognition effectiveness. The number of publications describing the forms of such subregions, the ways of obtaining and using them, today is no longer covered by references within the bibliography of this article – their infinite set is available on the Internet, by keywords, for example, "Diagonal PCA", "Matrix PCA", "Extension PCA", "Mixture of Bilateral-Projection 2DPCA".

Since 2005, algorithms for the two-dimensional projection into eigensubspaces (within the PCA framework) have been presented by other groups of authors, but already as "Generalized Low Rank Approximations of Matrices", "Generalized 2DPCA", and even "2DSVD" [19–21]. In these works, there is a reference to the article [17], which does not represent the ideas of the 2DPCA, but there is no reference either to [8] or to [9–16], which is rather strange for scientific publications.

As for the "2DSVD method" from [21], two important facts should be noted here: if the image database parameters are defined as $L = 1$ and $K = 1$, the 2DPCA/2DKLT method "degenerates" into the SVD method; if $K > 1$ and $L \geq 1$, the 2DPCA/2DKLT method does not become a 2DSVD method, since in general the covariance matrices computed in the 2DPCA methods have full rank and, therefore, are not singular. And this second fact does not allow to name 2DPCA/2DKLT, as 2DSVD. Incorrect use of the 2DSVD name in [21] arose from the association that the PCA can be implemented via SVD. Obviously, also that the solution of eigenvalue problems in 2DPCA can be realized through two independent SVDs, but this is not the same thing, since the entire 2DPCA/2DKLT algorithm includes not only eigenvalue problems.

It is interesting to note that the 2DSVD method is used mainly as a tool for processing terrain maps for their compression, encoding video data, processing handwritten texts for classification problems and numerical matrices for problems of their low-rank approximation [22–24]. The 2DPCA methods are traditionally used as a tool in the tasks of facial biometrics – the reduction of the input features space, the recognition of face images and the construction of their models. This will be discussed in more detail in the next section.

5. METHODS OF TWO-DIMENSIONAL PROJECTIONS FOR TWO SETS OF FACE IMAGES

It would seem very strange that the methods of two-dimensional projections into their eigensubspaces received the greatest development and application in face image processing tasks. However, this fact has many reasons, which are presented below. The peculiarity of face image (FI) processing is that various sensors of FI, various methods of preprocessing FI and various forms of their representation can be used within the corresponding systems. In general, FI can be represented in the form of 2D images in visible (VIS), thermal (NIR) and infrared light (IR), in the form of different sketches (forensic, composite, art, viewed sketch) and populations of them; in the form of a "range image" – form 2.5D, as in the form of contour models of the face area (Active Shape Model – ASM), models of the appearance (Active Appearance Model – AAM), defining the texture of IL, and finally in the form of 3D models of FI.

In this case, the input data at the input of the system can be simultaneously represented by several sets of FI connected "in pairs", "triples" or even "groups of more than 3 FI". An example of the last option is five sets of FI representing the face of the same person, of which the first set contains VIS FI, the second is NIR FI, the third IR FI, fourth FI in the form of sketches, and finally, the fifth set contains FI in the form of 3D. In biometrics, such sets are referred to as "multisensory and/or multimodal nature" or referred to the group of "heterogeneous data". The most common links in the technical literature within the framework of mutual transformations of FI are shown in Fig. 3.

On the one hand, the presence of a variety ways to represent FI significantly expands the capability and areas of face retrieval and face recognition systems application. On the other hand, this diversity considerably complicates the structure of the corresponding face recognition systems, the algorithms of their functioning and implementation of such systems. For example, in order to realize the face retrieval, to operate with such data and to interpret them, it is necessary to relate them to each other (for example, two images of different sensory or different semantic nature). Further, if it is assumed that there is a mutual correspondence between these data, then when querying for data entering one of the sets, it is easy to find the corresponding result from another set. The necessary mutual correspondence (like mutual indexing) is achieved in the training stages based on multisensory samples (or heterogeneous) data.

With this in mind, in recent years' new classes of tasks have been formed: "Heterogeneous Face Recognition and Matching", "Cross-Modal Face Matching", "Face Image Indexing and Retrieval", as well as more general approaches for searching information, for example, "Cross-Modal Multimedia Retrieval". The main feature of these tasks is working with two or more sets of FI (or other data forms).

The negative characteristic of all the above projection methods into their eigensubspaces was that they were initially focused on processing only one set of data, although the parameters $K \geq 1$ and $L \geq 1$ of these data could determine their complex structural organization. However, despite the fact that these methods do not correspond to modern classes of image processing tasks, some of their solutions have been obtained, both within the one-dimensional and two-dimensional methods of PCA and LDA. In this case, each separate set of source data was projected into a separate proper subspace, and then a mutual transformation of the images between these subspaces was used. For example, in [25] experiments were performed on the basis of 1DPCA and two sets of FI – a set of photos and a set of corresponding sketches [26]. At the same time, in [25] the possibility of constructing models of mutual reconstruction for photo-sketch pairs is shown, as well as solving the problem of recognizing photos and sketches with the help of such models. Later, on the ideas laid down in [25], it was possible to solve the superresolution problems between photos and sketches (often these tasks are referred to as

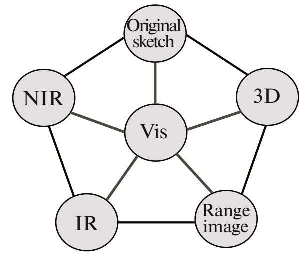


Fig. 3. Relations within the framework of mutual transformations of FI.

"hallucinations" in the application to FI) and problems of FI populations generation. So in [27] experiments were performed both on the basis of 1DPCA, and based on 2DPCA/2DKLT for two sets containing mixed groups of images. These groups include: photos and sketches, VIS and NIR images, photos and corresponding contour images, and images that are not semantically related in the corresponding pairs. The sources of these images are given in [27]. An example of the result of mutual reconstruction of images for such sets is shown in Fig. 4.

Here, in series 1 and 2, training sets of data are presented. Series 3 – the result of reconstruction, obtained from the images of series 2. Row 4 is a series 2 image in low resolution with the addition of noise. Series 5 – the result of reconstruction of FI from images of series 4. Here, in the learning process, the original data is transformed into its eigensubspaces, and also two matrices of mutual transformation of images are formed, which are further used for mutual transformation of images in pairs. The methods presented in [27] were also used to generate photo populations from given sketches. An example of these results is shown in Fig. 5.

And, nevertheless, the solution of the above-mentioned classes of problems is most fully realized only on CCA ideas (laid down already in the beginning of the 20th century [4]) and PLS regression ideas first implemented in econometrics and developed in chemometry. In application to the digital processing of multidimensional signals (but on the fact of images), PLS algorithms, for example, can be considered as "extending PCA/SVD methods to two or more sets of source data" [28].

The CCA and PLS methods allow us to display two sets of input data (often not correlated, or externally similar, but not semantically related) to a common subspace of features in which they highly correlate. This allows us to connect the pairs of input data in eigensubspace with the help of a common regression model to represent and understand certain observations, actions (or phenomena) through others, thereby eliminating, in whole or in part, the sensory or semantic gap between them.

Fig. 4. Training sets and the results of mutual reconstruction.

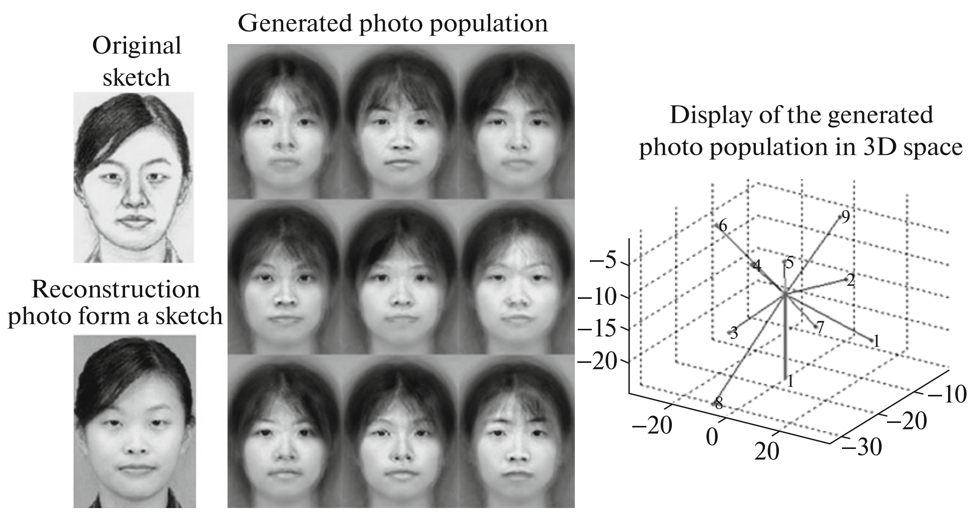


Fig. 5. An example of reconstructing a photo from a sketch that is not part of the training base, the result of generating a population of nine photos and displaying it in a 3D subspace on the first three features.

The first applications of CCA and PLS for face processing appeared at the beginning of this century. And the first were the following tasks: reconstruction of some FI for others (for example, in [29] reconstruction of the form **VIS(rgb)** $\rightarrow$ **3D** and **VIS(rgb)** $\rightarrow$ **NIR)**; indexing, recognition and reconstruction of FI in heterogeneous sets of FI (for example, in [30] the "Range Image" forms and the texture of color FI were used);

matching parameters of models of appearance for FI (Active Appearance Model [31]). It should be noted that at the initial stages of applying the methods of CCA and PLS, the tasks of processing heterogeneous (or multisensory) data were solved on the basis of one-dimensional algorithms implementation, which required the preliminary presentation of FI in the form of vectors. And here again there was a problem SSS, accompanying the tasks of FI processing. For its clarification, the main ideas of one-dimensional algorithms realization of CCA and PLS are considered in accordance with [28].

Let we are given two sets of input data, consisting of K number matrices, and each matrix represents FIs with size $M \times N$, where $MN \gg K$, which is usual for FI processing tasks. Each FI is "expanded into a vector" by concatenating its rows (or columns) and centering on the mean value in this vector. In this case, get two sets of input data X and Y, consisting of K vectors of size $MN \times 1$ each such that:

$$X = [X^{(1)}X^{(2)}\dots X^{(K)}]; \quad Y = [Y^{(1)}Y^{(2)}\dots Y^{(K)}], \quad \forall k = 1,2\dots K. \tag{7}$$

The goal of CCA is to find two projection matrices W_x and W_y, and transform the original data X and Y into a eigensubspace of canonical variables

$$U = W_x^T X, \quad V = W_y^T Y. \tag{8}$$

so that the following condition is satisfied: $\|U - V\| \to \min$.

The pairs of initial data $X^{(k)}$ and $Y^{(k)}$ cannot be connected with each other, while the canonical variables $U^{(k)}$ and $V^{(k)}$, defined as

$$U^{(k)} = W_x^T X^{(k)}, \quad V^{(k)} = W_y^T Y^{(k)}, \tag{9}$$

are connected by a stable correlation, the maximum of which is achieved when solving two joint eigenvalue problems [28]:

$$\left\{\begin{matrix}(C_{xx}^{-1}C_{xy}C_{yy}^{-1}C_{yx})W_x = \Lambda_x W_x \\ (C_{yy}^{-1}C_{yx}C_{xx}^{-1}C_{xy})W_y = \Lambda_y W_y\end{matrix}\right\}, \tag{10}$$

where: C_{xx}, C_{yy}, C_{xy}, C_{yx} – covariance matrices orders MN, and

$$C_{xx} = XX^T; \quad C_{yy} = YY^T; \quad C_{xy} = XY^T; \quad C_{yx} = C_{xy}^T;$$

Λ_x, Λ_y – diagonal eigenvalues matrices; W_x and W_y – eigenvectors matrices (projection matrices).

The goal of PLS is also to find two projection matrices W_x and W_y, and transform the original data into a eigensubspace so that the condition for maximizing the covariance between the variables in this subspace is fulfilled. In this case, the pairs of initial data $X^{(k)}$ and $Y^{(k)}$ can also not be connected, while the pairs of variables $U^{(k)}$ and $V^{(k)}$ in their eigesubspace are linked by a stable correlation, the maximum of which is achieved when solving the following two joint eigenvalue problems:

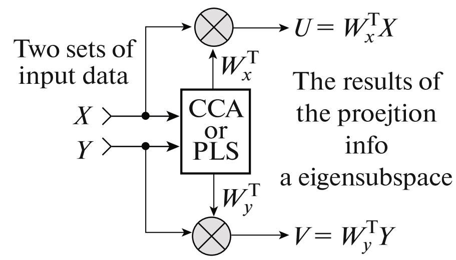


Fig. 6. Initial data and results of one-dimensional algorithms of CCA and PLS.

$$\begin{cases}(C_{xy}C_{yx})W_x = \Lambda_x W_x; \\ (C_{yx}C_{xy})W_y = \Lambda_y W_y,\end{cases} \tag{11}$$

where the matrices of mutual covariance are defined similarly to (10).

The initial data, the computation structure in CCA and PLS, as well as the results of U and V, can be schematically represented as shown in Fig. 6, where the blocks "⊗" implement the procedure for projecting the original data into a eigensubspace of variables.

Calculations in CCA and PLS result in the choice of the value of the parameter d – the reduction in the dimension of the feature space (with $d < MN$). Then the one-dimensional Karunen-Loeve transformation (KLT) is realized, as a result of which a complete representation of the variables U and V in the eigensubspace or their representation only in the principal components are obtained. In the latter case, KLT is realized using only those eigenfunctions that correspond to the principal components.

We draw attention to the relation (10), in which the common scattering matrices are obtained as a result of multiplying four symmetric (by definition) matrices of orders MN. Two of them – the matrix of autocovariance – require prior inversion, that for matrices formed by FI (and under the conditions $MN \gg K$), it is not trivial, since the rank of these matrices may be less than their order. The common scattering matrices also have orders MN, but they are not symmetric. The solution of the generalized eigenvalue problem in this case will be either unstable or not possible in principle even at $M > 100$ and $N > 100$ (although, for example, in biometric standards, FI of size 320 × 240 is used). Obviously, these characteristics limited the wide use of one-dimensional CCA methods in image processing tasks.

Two approaches for implementing CCA and PLS using SVD procedures are presented in [32, 33].

In [32] the solution of the task of tracking the face area in the process of forming the 3D model of the FI is considered. Here the area of interest on FI is not represented by all MN luminance features, but by a small (≤100) set of anthropometric coordinates on the face area. In this case, it is possible to realize CCA using three SVD procedures, it is possible to avoid the calculation of the original covariance matrices from (10), the matrix inversions required here, and also do without solving the eigenvalue problems.

In [33], the solution of the problems of cross-modal recognition of FI having a different sensory nature, such as: sketches, photo-originals, high and low resolution FI and different pose, etc., is considered. However, in this case, CCA via SVD can be applied only for a small number of pairs of FI, and the size of the images is also constrained. The latter is due to the fact that the orders of orthogonal matrices formed within the SVD are determined by the values of MN and K from expression (7). In addition, all solutions in this approach can be realized only in variants of projections in one direction, which requires the representation of the original pairs of FI in the form (7).

The "revolution" in solving the problems of processing two sets of FI began in 2007 with the appearance of the first articles on 2DCCA methods [34, 35].

An iterative-cascade algorithm for computing two projection matrices was presented in [34]. The main drawback of these solutions is the complexity of understanding and implementing the iterative-cascading 2DCCA algorithm in the conditions of a priori choice of the number of iterations, the convergence parameter and subsequent assurance of its convergence. And, although the authors note a significant decrease in computational costs in the framework of 2DCCA (compared to 1DCCA), in their experiments the size of FI did not exceed 50 × 50. And this also indirectly emphasizes the complexity of implementing and applying the iteration-cascading 2DCCA algorithm.

In [35], the implementation of the "2DCCA" method, based on the ideas of [17], was presented. And, of course, all the features (or rather the shortcomings) from [17] went to this method: "2DCCA" is implemented in one direction here, while in the other direction only the size of the original image is reduced. But even in this approach (when performing the experiments), the input FI had to be reduced in size by 28 × 23. Which also indicates the imperfection of this approach.

Despite the noted imperfection of the approaches [34, 35], they determined two independent branches of the CCA application in the FI processing. Although in publications based on these approaches, often the parameters of the CCA implementation (and/or the sizes of the initial and output data) in the experiments are unclear.

The first non-iterative 2DCCA/2DKLT algorithm, the continuing approach [8–11] and based on the 1DCCA algorithm from [28], was presented in [36] in 2009. In the 2DCCA/2DKLT record, two stages of execution are emphasized: analysis of the input data in two directions (by columns and rows of FI) with the choice of the principal components and the two-dimensional Karunen-Loeve transformation. Ideas [36], as well as the idea of the non-iterative algorithm 2DPLS/2DKLT [37–42] are presented below in the form of pseudocodes.

6. NON-ITERATIVE IMPLEMENTATION ALGORITHMS FOR 2DCCA/2DKLT AND 2DPLS/2DKLT

Let X and Y be two sets of input data, consisting of K matrices of size $M \times N$, where $MN \gg K$:

$$\begin{aligned} X &= [X^{(1)} X^{(2)} \dots X^{(K)}] \\ \text{and} \quad Y &= [Y^{(1)} Y^{(2)} \dots Y^{(K)}], \quad \forall k = 1, 2, \dots K. \end{aligned} \tag{12}$$

Matrices $X^{(k)}$, $Y^{(k)}$ form related pairs – as, for example, FI of one person, but obtained from different sensory sources.

The goal of the 2DCCA/2DKLT and 2DPLS/2DKLT algorithms is to find the two pairs of matrices $\{W_{x1}, W_{y1}\}$ and $\{W_{x2}, W_{y2}\}$, used as left and right projection matrices in the realization of the two-dimensional Karunen-Loev transformation of the original data $X^{(k)}$ and $Y^{(k)}$, $\forall k$. This achieves their projection to the common eigenspace $X^{(k)} \rightarrow U^{(k)}$ and $Y^{(k)} \rightarrow V^{(k)}$ so that the following condition is satisfied:

$$\|U^{(k)} - V^{(k)}\| \rightarrow \min, \tag{13}$$

which was defined in [4] as a criterion for the correlation maximum between the variables in the eigenspace:

$$\underset{\{W_{x1}, W_{x2}, W_{y1}, W_{y2}\}}{\operatorname{argmax}} = \operatorname{Cov}(U, V). \tag{14}$$

In general, the original data $X^{(k)}$ and $Y^{(k)}$ may not be related, while the canonical variables $U^{(k)}$ and $V^{(k)}$ are correlated by a stable correlation. It is this property of CCA and PLS that creates the basis for the solution of the tasks of "Heterogeneous Face Recognition and Matching", "Cross-Modal Face Matching", "Face Image Indexing and Retrieval" tasks "Face Hallucination" and "Face Super-resolution" and even "Cross-Modal Multimedia Retrieval".

Versions of the 2DCCA/2DKLT and 2DPLS/2DKLT algorithms written in the form of pseudocodes are presented in Table 3.

Table 3. Pseudo-codes for the 2DCCA/2DKLT and 2DPLS/2DKLT

Algorithm **2DCCA/2DKLT**	Algorithm **2DPLS/2DKLT**
Input: $\{X^{(k)}, Y^{(k)}\} \in \Re^{M\times N}$, $\forall k = 1, 2, \ldots K$	**Input:** $\{X^{(k)}, Y^{(k)}\} \in \Re^{M\times N}$, $\forall k = 1, 2, \ldots K$
Output: $\{\bar{X}, \bar{Y}\}$, $\{\Lambda x_1, \Lambda x_2, \Lambda y_1, \Lambda y_2\}$, $\{Wx_1, Wx_2, Wy_1, Wy_2, U^{(k)}, V^{(k)}\}, \forall k$	**Output:** $\{\bar{X}, \bar{Y}\}$, $\{\Lambda x_1, \Lambda x_2, \Lambda y_1, \Lambda y_2\}$, $\{Wx_1, Wx_2, Wy_1, Wy_2, U^{(k)}, V^{(k)}\}, \forall k$.
1. Calculate the average images from the data: $\bar{X} = \frac{1}{K}\sum_{k=1}^{K} X^{(k)}; \quad \bar{Y} = \frac{1}{K}\sum_{k=1}^{K} Y^{(k)}$	1. Calculate the average images from the data: $\bar{X} = \frac{1}{K}\sum_{k=1}^{K} X^{(k)}; \quad \bar{Y} = \frac{1}{K}\sum_{k=1}^{K} Y^{(k)}$
2. Center the source data $\forall k$: $\bar{X}^{(k)} = X^{(k)} - \bar{X}; \bar{Y}^{(k)} = Y^{(k)} - \bar{Y}$.	2. Center the source data $\forall k$: $\bar{X}^{(k)} = X^{(k)} - \bar{X}; \quad \bar{Y}^{(k)} = Y^{(k)} - \bar{Y}$.
3. Calculate the covariance matrices with respect to rows of matrices in p.2: $C_{xx}^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{X}^{(k)}(\bar{X}^{(k)})^{\mathrm{T}}; \quad C_{yy}^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{Y}^{(k)}(\bar{Y}^{(k)})^{\mathrm{T}};$ $C_{xy}^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{X}^{(k)}(\bar{Y}^{(k)})^{\mathrm{T}}; \quad C_{yx}^{(\mathrm{r})} = (C_{xy}^{(\mathrm{r})})^{\mathrm{T}}.$	3. Calculate the covariance matrices with respect to rows of matrices in p.2: $C_{xx}^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{X}^{(k)}(\bar{X}^{(k)})^{\mathrm{T}}; \quad C_{yy}^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{Y}^{(k)}(\bar{Y}^{(k)})^{\mathrm{T}};$ $C_{xy}^{(\mathrm{r})} = \sum_{k=1}^{K} \bar{X}^{(k)}(\bar{Y}^{(k)})^{\mathrm{T}}; \quad C_{yx}^{(\mathrm{r})} = (C_{xy}^{(\mathrm{r})})^{\mathrm{T}}.$
4. Calculate the covariance matrices with respect to columns of matrices in p.2: $C_{xy}^{(\mathrm{c})} = \sum_{k=1}^{K} (\bar{X}^{(k)})^{\mathrm{T}}\bar{Y}^{(k)}; \quad C_{yx}^{(\mathrm{c})} = (C_{xy}^{(\mathrm{c})})^{\mathrm{T}}.$ $C_{xx}^{(\mathrm{c})} = \sum_{k=1}^{K} (\bar{X}^{(k)})^{\mathrm{T}}\bar{X}^{(k)};$ $C_{yy}^{(\mathrm{c})} = \sum_{k=1}^{K} (\bar{Y}^{(k)})^{\mathrm{T}}\bar{Y}^{(k)};$	4. Calculate the covariance matrices with respect to columns of matrices in p.2: $C_{xy}^{(\mathrm{c})} = \sum_{k=1}^{K} (\bar{X}^{(k)})^{\mathrm{T}}\bar{Y}^{(k)};$ $C_{yx}^{(\mathrm{c})} = (C_{xy}^{(\mathrm{c})})^{\mathrm{T}}.$
5. Compute the common scattering matrices: $\begin{cases} S^{(1,\mathrm{r})} = [C_{xx}^{(\mathrm{r})}]^{-1}C_{xy}^{(\mathrm{r})}[C_{yy}^{(\mathrm{r})}]^{-1}C_{yx}^{(\mathrm{r})}; \\ S^{(2,\mathrm{r})} = [C_{yy}^{(\mathrm{r})}]^{-1}C_{yx}^{(\mathrm{r})}[C_{xx}^{(\mathrm{r})}]^{-1}C_{xy}^{(\mathrm{r})}; \end{cases}$ $\begin{cases} S^{(1,\mathrm{c})} = [C_{xx}^{(\mathrm{c})}]^{-1}C_{xy}^{(\mathrm{c})}[C_{yy}^{(\mathrm{c})}]^{-1}C_{yx}^{(\mathrm{c})}; \\ S^{(2,\mathrm{c})} = [C_{yy}^{(\mathrm{c})}]^{-1}C_{yx}^{(\mathrm{c})}[C_{xx}^{(\mathrm{c})}]^{-1}C_{xy}^{(\mathrm{c})}; \end{cases}$	5. Compute the common scattering matrices: $\begin{cases} S^{(1,\mathrm{r})} = C_{xy}^{(\mathrm{r})}C_{yx}^{(\mathrm{r})}; \\ S^{(2,\mathrm{r})} = C_{yx}^{(\mathrm{r})}C_{xy}^{(\mathrm{r})}; \end{cases}$ $\begin{cases} S^{(1,\mathrm{c})} = C_{xy}^{(\mathrm{c})}C_{yx}^{(\mathrm{c})}; \\ S^{(2,\mathrm{c})} = C_{yx}^{(\mathrm{c})}C_{xy}^{(\mathrm{c})} \end{cases}$
6. Solve 4 problems with eigenvalues: $\begin{cases} S^{(1,\mathrm{r})}W_{x_1} = \Lambda_{x_1}W_{x_1}; \\ S^{(2,\mathrm{r})}W_{y_1} = \Lambda_{y_1}W_{y_1}; \end{cases} \quad \begin{cases} S^{(1,\mathrm{c})}W_{x_2} = \Lambda_{x_2}W_{x_2}; \\ S^{(2,\mathrm{c})}W_{y_2} = \Lambda_{y_2}W_{y_2} \end{cases}$ and determine the eigenvalues matrices (Λ_{*1} и Λ_{*2}) and the eigenvectors matrix, and in fact the left (W_{x1}, W_{y1}) and right (W_{x2}, W_{y2}) projection matrices.	6. Solve 4 problems with eigenvalues: $\begin{cases} S^{(1,\mathrm{r})}W_{x_1} = \Lambda_{x_1}W_{x_1}; \\ S^{(2,\mathrm{r})}W_{y_1} = \Lambda_{y_1}W_{y_1}; \end{cases} \quad \begin{cases} S^{(1,\mathrm{c})}W_{x_2} = \Lambda_{x_2}W_{x_2}; \\ S^{(2,\mathrm{c})}W_{y_2} = \Lambda_{y_2}W_{y_2} \end{cases}$ and determine the eigenvalues matrices (Λ_{*1} и Λ_{*2}) and the eigenvectors matrix, and in fact the left (W_{x1}, W_{y1}) and right (W_{x2}, W_{y2}) projection matrices.
7. To project the original data into eigenspace, realizing the two-dimensional Karunen-Loeve transformation: $U^{(k)} = W_{x_1}^{\mathrm{T}}\bar{X}^{(k)}W_{x_2}, \quad \forall k$ $V^{(k)} = W_{y_1}^{\mathrm{T}}\bar{Y}^{(k)}W_{y_2}, \quad \forall k$	7. To project the original data into eigenspace, realizing the two-dimensional Karunen-Loeve transformation: $U^{(k)} = W_{x_1}^{\mathrm{T}}\bar{X}^{(k)}W_{x_2}, \quad \forall k$ $V^{(k)} = W_{y_1}^{\mathrm{T}}\bar{Y}^{(k)}W_{y_2}, \quad \forall k$
8. To reconstruct the initial data, perform the inverse two-dimensional Karunen-Loeve transformation: $\bar{X}^{(k)} = W_{x_1}U^{(k)}W_{x_2}^{\mathrm{T}}, \quad \forall k;$ $\bar{Y}^{(k)} = W_{y_1}V^{(k)}W_{y_2}^{\mathrm{T}}, \quad \forall k.$	8. To reconstruct the initial data, perform the inverse two-dimensional Karunen-Loeve transformation: $\bar{X}^{(k)} = W_{x_1}U^{(k)}W_{x_2}^{\mathrm{T}}, \quad \forall k;$ $\bar{Y}^{(k)} = W_{y_1}V^{(k)}W_{y_2}^{\mathrm{T}}, \quad \forall k.$
End of algorithm	**End of algorithm**

The dimensions of all matrices calculated above	The dimensions of all matrices calculated above
$\{\bar{X}^{(k)}, \bar{Y}^{(k)}\} \in \Re^{M\times N}$;	$\{\bar{X}^{(k)}, \bar{Y}^{(k)}\} \in \Re^{M\times N}$;
$C^{(\mathrm{r})} \in \mathrm{R}^{M\times M}$; $C^{(\mathrm{c})} \in \Re^{N\times N}$;	$C^{(\mathrm{r})} \in \mathrm{R}^{M\times M}$; $C^{(\mathrm{c})} \in \Re^{N\times N}$;
$\{W_{x_1}, W_{y_1}, \Lambda_{*1}\} \in \Re^{M\times M}$;	$\{W_{x_1}, W_{y_1}, \Lambda_{*1}\} \in \Re^{M\times M}$;
$\{W_{x_2}, W_{y_2}\Lambda_{*2}\} \in \Re^{N\times N}$.	$\{W_{x_2}, W_{y_2}\Lambda_{*2}\} \in \Re^{N\times N}$.
Matrices: C – symmetric;	Matrices: C – symmetric;
Λ_{*1}, Λ_{*2} – are diagonal;	Λ_{*1}, Λ_{*2} – are diagonal;
W_{*1}, W_{*2} – "almost orthogonal."	W_{*1}, W_{*2} – orthogonal.

The reduction of original feature space dimension is realized as a "truncated 2DKLT", in which only eigenvectors that correspond to the d principal components participate in further transformations. For this, from the left projection matrices $\{W_{x1}^T, W_{y1}^T\}$ we select d rows, and from the right matrices of the projection $\{W_{x2}, W_{y2}\}$ we select d columns corresponding to d to the largest eigenvalues, and on their basis we form four reduction matrices $\{Fx_1, Fx_2\}$ and $\{Fy_1, Fy_2\}$. Wherein $d < \min(M, N)$ or $d_1 < M$; $d_2 < N$, if $d_1 \neq d_2$. The upper bounds of the parameter d are determined based, for example, on the energy significance of the eigenvalues, and the lower bounds are chosen taking into account the desired quality of the approximation of the original data.

The truncated two-dimensional Karunen-Loew transformation is realized as follows:

$$\hat{U}^{(k)} = F_{x1}X^{(k)}F_{x2}; \quad \hat{V}^{(k)} = F_{y1}Y^{(k)}F_{y2}; \quad \forall k. \tag{13}$$

The sign "^" determines the difference between the result of the approximation and the result of the reconstruction $U^{(k)}$ and $V^{(k)}$. The matrices $\{Fx_1, Fy_1\}$ and $\{Fx_2, Fy_2\}$ have dimensions $(d \times M)$ and $(N \times d)$, respectively, or $(d_1 \times M)$ and $(N \times d_2)$ in the general case. In this case, result (5) – a matrix of size $d \times d$ (or $d_1 \times d_2$) represents original images in the eigensubspace of characteristics.

7. PECULIARITY OF PRACTICAL IMPLEMENTATION OF CCA AND PLS FOR FACE IMAGES

a) FI, being data received from various sources, can contain both data omissions (regions consisting of zero values for rows and columns) and regions with duplicate (equal in value) data. In this case, the rank of the covariance matrices (clause 4 of the above pseudocode) for $K < 10$ may be less than their order. To eliminate this effect, it is sufficient to impose noise on the original images with amplitude ≤5% of the maximum over the used range of image brightness. Sometimes, in 2DCCA algorithm, instead of centering the source images X and Y, it is necessary to perform their normalization (for example, $\hat{X}^{(k)} = X^{(k)}/\|X\|$). In this case, the norm of each image in the sets X and Y will be equal to "1", and the average value is close to 1, which will correspond to their "bleaching".

b) As shown in clause 5 of the above pseudocode, the calculation of common scattering matrices in the 2DCCA algorithm uses the inversion operations of the autocovariance matrices, which may be poorly conditioned. Therefore, before the inversion of these matrices, it is necessary to regularize them.

c) The common scattering matrices S in 2DCCA algorithm must also be regularized, in view of their non-symmetry. This greatly simplifies the solution of the eigenvalue (EV) problems.

d) In theory, 2DCCA and 2DPLS algorithms require solving four tasks on EV (see point 6 of the pseudocode). However, both in 2DCCA/2DKLT algorithm and in the 2DPLS/2DKLT algorithm, it is possible to reduce the number of solved tasks on EV to two (!).

We show how to do this for 2DCCA/2DKLT algorithm. Consider the left-hand pair of tasks on the EV (in point 6 of the pseudocode), which is defined relative to the rows of the images of the sets X and Y. Solve the first problem on EV and find the projection matrix W_{x1}. The second matrix W_{y1} is computed without solving the problem on EV as follows [28]:

$$W_{y1} = [C_{yy}^{(\mathrm{r})}]^{-1} C_{yx}^{(\mathrm{r})} W_{x1}. \tag{14}$$

And then the calculated eigenvectors (that is, the columns of the matrix W_{y1}) must be normalized to 1, dividing them into a "proper norm". You should also do it for the second (right) pair of tasks on the EV – that is, solve one task on the EV to obtain the projection matrix W_{x2}, and recalculate the matrix W_{y2} from it.

For algorithm 2DPLS/2DKLT we do the same, but without using the autocovariance matrices in the product (14). The justification and elucidation of these solutions can be found, for example, in [28, 41].

And now we list and estimate the computational costs for the remaining operations that must be performed when implementing the parallel 2DCCA algorithm for X and Y image sets with the parameters $\{M, N, K\}$, where $MN \gg K$ and when 2DCCA is performed on all the brightness features:

1) formation of six covariance matrices: $\boldsymbol{O}(3K(M^2N+N^2M))$;

2) inversion of 4 matrices of autocovariation: $\boldsymbol{O}(2(M^3+N^3))$;

3) calculation of the projection matrix W_{x1} (solution of the EV problem): $\boldsymbol{O}(M^3)$;

4) formation of the matrix W_{y1} according to (14) together with its normalization: $\boldsymbol{O}(2M^3+M^2)$;

5) calculation of the projection matrix W_{x2} (solution of the EV problem): $\boldsymbol{O}(N^3)$;

6) formation of the matrix W_{y2} according to (14) together with its normalization: $\boldsymbol{O}(2N^3+N^2)$.

Then the total costs will be no more than $\boldsymbol{O}(3K(M^2N+N^2M)+5(M^3+N^3)+M^2+N^2)$.

If we assume that $M=N$ и $M^3 \gg M^2$, then we can write:

$$3K(M^3+M^3)+5(M^3+M^3)=6KM^3+10M^3. \quad (15)$$

Now, if $K>10$, then the main computational costs are attributed to the formation of six covariance matrices. Their calculation is based on multiplying the matrices $X^{(k)}$ and $Y^{(k)}$ and summing the result $\forall k$. We note that in the iteration algorithms of 2DCCA, six covariance matrices are also generated over the entire set of initial data and also in the parameter of the number of iterations. In this case, each covariance matrix is based on the multiplication of 4-number matrices, of which two represent the original data, and the other two are the left and right projection matrices formed during iteration. The computational costs of the iterative algorithms obtained in this case will be even greater than that shown in (15). Therefore, if the original data are not the luminance features of the original images, but the result of any transformation (gradient, spectral, by principal components (via PCA, for example)) so that $M \to m$ and $N \to n$, and $m \ll M$, a $n \ll N$, then the costs will be significantly reduced. This is what we see when we use the iterative algorithms of 2DCCA in application to the FI (reduction of the sizes of the original images, a transition to another feature space, etc.). But here we must remember that such a transformation of FI is not always possible and not in all scenarios is permissible.

Instead of the iterative algorithms 2DPCA/2DKLT and 2DPLS/2DKLT work both in the holistic representation of FI, and their preliminary transformation into an intermediate subspace features. And the simple calculations implemented in them are easier to program, and perform with hardware support! Thus, in Fig. 7 shows a parallel and cascade form of the implementation of the algorithms 2DCCA/2DKLT and 2DPLS/2DKLT. The structure of the parallel algorithm corresponds to the description of the pseudocode given above. The structure of the cascade algorithm consists of two cascades, shown in Fig. 6, and each cascade implements 1DCCA (or 1DPLS). In this case, the first cascade implements CCA algorithm (or PLS) relative to the rows of the original data $X^{(k)}$ and $Y^{(k)}$, and the second cascade is relative to the columns of the result obtained in the first cascade.

8. EXAMPLES OF USING CCA AND PLS FOR PROCESSING SETS OF FACE IMAGES

Our own results of applying 2DCCA/2DKLT and 2DPLS/2DKLT in the problems of processing FI are given in the articles [37–42]. In these papers the following results are presented: a comparative analysis of the solution of the problems of recognition of FI (of different sensory and gender nature); mutual indexing and recognition of FI and other multimedia objects not connected semantically; specialty of the parallel and cascade schemes of 2DCCA/2DKLT and 2DPLS/2DKLT implementation with an estimation of their characteristics; the systems of recognition of FI with various variants of fusion of signs in eigen subspace and various types of classifiers are determined and investigated. Also, comparative results of solving the same problems are presented in the framework of other methods known in the practice of FI recognition. The experiments presented in these works were performed on the following bases: the FERET base, "Family album – FA" database, "Equinox" database,

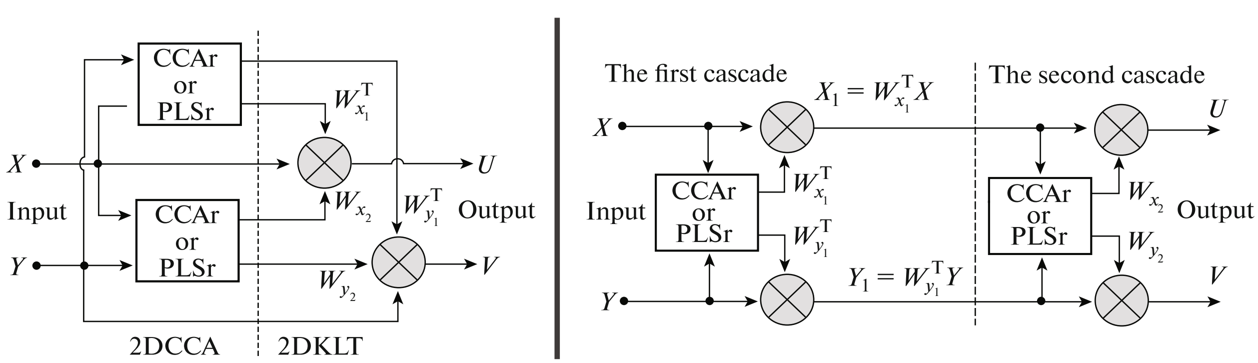


Fig. 7. The structure of the parallel and cascade algorithms for 2DCCA/2DKLT and 2DPLS/2DKLT implementation.

Fig. 8. Images of base "People and Dogs."

Fig. 9. Input images and the results of their mutual reconstruction.

ZUT (VIS/NIR) database, "People and dogs" database.

For example, FA base is made up of a set of male FI (husband) and a set of female FI (women), whose source was FERET benchmark database. Each FI of the FA base had a size of 224 × 184 and did not change in size, nor did it improve in quality throughout the experiment, although the brightness of the images in pairs changed significantly, which is a feature of the FERET database. Each connected pair of the FI of the FA base, including people of different sexes, in fact contained semantically unrelated data. Taking into account this fact, and also taking into account the fact that the pairs of FI were selected at random, the correlation between them in the original feature space was close to zero. At the same time, the mutual correlation in its eigensubspace was much higher than 0.5, and in some classes it reached values >0.85. Within the "husband-wife" mutual recognition experiments the average results were: not less than 80% for the CCA method and 89% for the PLS method with the number of principal components $d \leq 20$.

In "Equinox" database, the pairs connected with the FI included persons obtained in visible light and infrared radiation (VIS and Th). The sizes of the initial ILs were 320 × 240 and were not changed. It should be noted that the quality of FI/Th was not high enough because of the many attractors whose appearance is related to its production method, as well as the non-coincidence of VIS and Th FI sizes and the presence of "empty" lines. In our experiments, the quality of these images did not improve. Taking into account the noted, the correlation between VIS and Th FI in the source feature space was also close to zero. Within the framework of performed VIS ↔ Th FI mutual recognition experiments, the average results were: not less than 92% for both methods with characteristic sizes in the subspace of not more than 20 × 20.

"People and Dogs" database [43] represents the case where the source images do not belong to the same global class (see Fig. 8). Pairs of images in classes have some external similarity (between the FI – "owner" and his dog), which is determined by the following factors: a close-in expression of the face of the owner and "muzzle" of the dog; the same foreshortening of two portraits; close in form to the owner's hairdo and the exterior of the dog; color gamut – the same color of the hair of the host and the coloration of the dog's coat, as well as the texture.

Within this framework, we set the following goals: to check whether the subjective evaluation of the similarity of these images is comparable with the formal estimates of their similarity; which methods of formal similarity estimation can be considered reliable; what is the measure of the similarity of these images in the space of canonical variables.

The answer to the last question would prove the possibility of using 2DCCA methods in problems of searching for a connection between loosely coupled data. It turned out that the highest correlation value for brightness histograms in pairs was 0.8 (which corresponded to subjective perception). However, the phase correlation cov($X^{(k)}$, $Y^{(k)}$), estimating the texture and shape of the objects in the images was close to "0", and the highest value of the similarity level in pairs according to the **S**tructural **SIM**ilarity **I**ndex (SSIM) was 0.28, which is reflected over the pairs of images (Fig. 8).

At the same time, in the space of canonical variables, the value of the phase correlation cov (U, V) for this base surpassed the level of 0.6, which allowed successfully solving the problem of mutual indexing of these images in the corresponding pairs, as well as the problem of mutual recognition of FI by dog images and vice versa. These results also formed the basis for constructing algorithms for mutual reconstruction of some images by others using the regression between sets of variables in the eigensubspace. Examples of such reconstruction under conditions of "extremely small sample" ($MN \gg K$) are shown in Fig. 9 for training sets X and Y, consisting of only three images, each measuring 200 by 150 pixels.

Here: columns with number 1 – a pair of images $X^{(k)}$ and $Y^{(k)}$ from training sets; columns with number 2 – the results of mutual reconstruction (dog $\leftrightarrow$ FI) for regression calculated in its own pair; columns with number 3 – the results of mutual reconstruction (dog $\leftrightarrow$ FI) by regression calculated according to training data.

Regression in both cases is constructed in the characteristic space of canonical variables, and the formation of the reconstruction result is realized on the basis of the inverse 2DKLT from the result of the corresponding regression.

Thus, in works [41, 42] the possibility of using 2DCCA/2DKLT methods in problems of establishing a connection between semantically unrelated data was shown, as well as the possibility of their mutual indexing, recognition and reconstruction. At the same time, the possibility of solving such problems in the conditions of small sample (Small Sample Size) was experimentally confirmed, since in the above experiment, for example, $MN = 30000$, and $K = 3$.

Of the other original applications for 2DCCA and 2DPLS, we can mention [44, 45].

In [44], the algorithm for solving the problem of super resolution of FI – reconstruction of high resolution (HR) FI from low resolution (LR) FI is presented. The training is realized within the framework of 2DCCA on sets of images X with high (HR) and Y with low resolution (LR), and the reconstruction of LR $\to$ HR (performed, as usual, in the subspace of canonical variables) is supplemented with the procedure for improving high-frequency components of the result. The experiments were carried out on two image bases and in both cases showed better results than the known (cited in the article) approaches. However, 2DCCA here was implemented using an iterative algorithm [34], which required to limit the size of the initial data and, additionally, required the decomposition of the original images into three parts. As a result, the iterative learning algorithm within 2DCCA, the reconstruction of LR $\to$ HR, the matching of HR result and the compensation of high-frequency components were performed here three times in sets of decomposed parts of images. Nevertheless, approach [44] for the problems of super resolution of FI is of practical interest, as methodically worked out and sufficiently clearly presented in the article, therefore it would be interesting to realize the ideas [44] also in the framework of non-iterative 2D CCA/2D KLT algorithms, not requiring reducing the size of the FI and their decomposition into parts.

In work [45] the solution of problems palm prints images recognition (Palmprint) and FI of base FERET is presented. The authors evaluate the computational complexity of the iterative approach to the implementation of 2DPLS in comparison with the non-iterative approach to the implementation of 2DPLS (with reference to its own work of 2005). In this case, the superiority of the iterative approach is shown in cases of preliminary transformation of the original images into a set of new features, the dimension of which is smaller than the original space of the brightness characteristics of the images. The superiority (in terms of accuracy and complexity of calculations) of 2DPLS approach in comparison with 1DPLS is also experimentally proved.

The overall result of this section is as follows:

—the ideas presented on the implementation of 2DCCA methods [34–36] identified three independent branches of CCA application in the treatment of FI;

—2DPLS algorithms evolved independently of the 2DCCA methods, often outperforming their ideas with the implementation of 2DCCA. One of the "examples of such advancement" is the independent branch [34–36] of the realization of PLS and CCA based on SVD (see, for example, the early works of Magnus Borga and David Weenink);

—presented in the section ideas and methods of implementation of 2DCCA and 2DPLS were applied to the tasks of "Heterogeneous Face Recognition and Matching", "Cross-Modal Face Matching", "Face Image Indexing and Retrieval" and "Face Hallucination", "Face Super-resolution" tasks, as well as the tasks of "Cross-Modal Multimedia Retrieval".

9. APPLICATION OF PROJECTION METHODS TO SUBSPACES IN NEURAL NETWORKS

At the borders of the 20th and early 21st centuries, a methodological (and technological) revolution started in computer training and computer vision, started by convolutional neural networks (Convolutional Neural Network – CNN). In image processing tasks (classification, recognition, coding and reconstruction, etc.), CNN works with original (unprocessed) data, so each pixel of the original image must be associated with an independent CNN input channel. With the growth in the size of the original images and the growing number of training samples (reaching up to several million images), processing such an image requires a large number of CNN layers. And it is connected, on the one hand, with an unnecessary information excess in the source images, and on the other hand – with an overabundance of similar information in the training data. With a very large number of layers, such networks are also referred to as deep neural networks (Deep NN). So, for example, in 2016 the number of layers of Deep NN exceeded 150! At the same time, the more layers contain Deep NN, the longer and harder it is to train. Also, with a large number of layers, it is possible to get into retraining (!), when the network loses its ability to generalize and does not recognize hidden processes and important information in the training sample, or, conversely, accept noise in the source data for important information.

That is why the researchers and users of Deep NN were forced to solve the problem of reducing the amount of initial data and the volume of the training sample, provided that important information is stored in them. This can be achieved, for example, by reducing the dimension of the feature space in the original images and/or by preliminary clearing them of noise, by presenting the original images only in the principal components. A decrease in the volume of training data can be achieved, for example, by indexing them in the training sample, by preliminary classification, and also by the presentation of data on the principal components. And these problems are well solved in the framework of the methods of projecting images into their eigensubspaces (PIES), which was shown above.

Further, several publications from recent years [46–51] will be presented, which show the use of PIES methods in Deep NN.

In one of the first survey papers [46], various NN variants, constructed with the inclusion of linear and nonlinear processing procedures based on one-dimensional and two-dimensional methods of SVD, PCA, LDA and CCA were considered, and also consider algorithms for their learning for these network options. The authors note that such NNs are easily implemented and effectively used in solving problems of adaptive image processing, blind separation of data (useful information from noise), pattern recognition and compression. And the introduction of these procedures in NN reduces the amount of information at their entrance, leaving only useful information.

In [47], a very simple Deep NN training procedure for image classification based on the PIES/PCA method and called PCANet is proposed. In this case, PCANet will include the following basic processing components: a filter bank implemented in a cascade of PCA procedures; binary hashing procedure; calculation of block histograms. It is noted that combining the methods of PIES with the methods of binary hashing, images histogram representation and pyramidal combination of the intermediate result in layers makes it possible to simplify and accelerate Deep NN training, and also to reduce the number of layers in them.

The article [48] proposes a version of Deep NN, called by authors as "Multiple scales PCANet", intended for recognition of FI. The network uses in-depth training on multiple scales from the sets of FI representations realized using PCA in the first two layers of a convolutional neural network. In fact, at this level, a population of new representations of the original data is formed, which increases their representativeness and leads to an improvement of the recognition problem. When building and teaching the network, the authors proceed from the assumption that the higher the level of the characteristic space (read – the larger the population), the more accurately and fully it can represent the semantics of the original data. The remaining part of NN is constructed in a manner similar to that considered in [47]. The article is accompanied by unique experiments simulating the work "Multiple scales PCANet" performed on several new benchmark databases of FI and, including, on the bases of FI (2D and 3D) obtained in uncontrolled conditions and various scenarios. The results obtained are compared with those known for this topic. In conclusion, it is noted that "Multiple scales PCANet" can be used in other tasks of computer vision – emotions recognition and tracking and will also show its high efficiency.

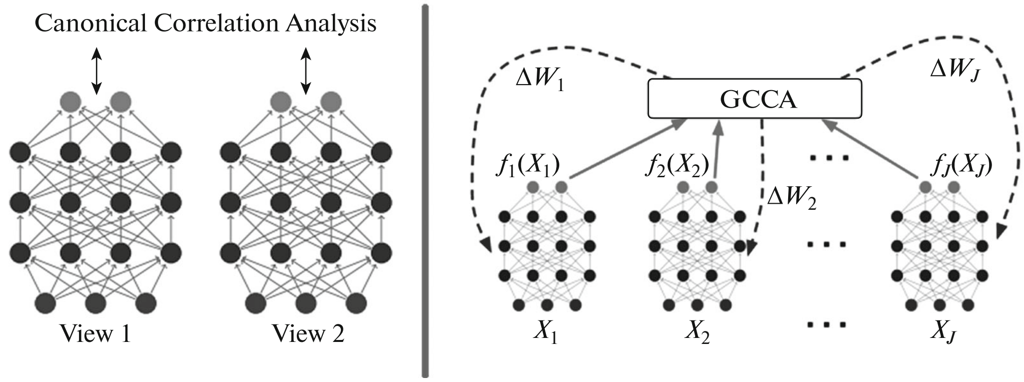


Fig. 10. Architecture of "Deep Canonical Correlation Analysis" [50] and "Deep generalized canonical correlation analysis" [51].

In [49], the combined architecture of Deep NN is proposed, in which the PCANet architecture [47] is combined with the possibilities of PLS regression ideas, which gives a new technique for classifying images, which the authors called "the PLSNet method". Here, in the first layers of NN, the reduction in the dimension of the feature space is realized by PCANet architecture, and in the next layers within PLS regression framework. Developing this idea further, the authors apply new filters to extract features in the next layers of NN and get a new Deep NN architecture, named by them as "*Improved* PLSNet". At the same time, higher accuracy of image classification is achieved than in PCANet. The experiments were performed on MNIST data sets (hand-written numbers) and CIFAR-10 (image sets).

In previous solutions, Deep NN's foundation is created by the procedure for reducing the feature space used at the entrance – in the first layers of the Deep NN. Articles [50, 51] consider new Deep NN architectures, called "Deep Canonical Correlation Analysis" and "Deep generalized canonical correlation analysis". Here, each set of input data (two sets in the architecture [50] and several sets in the architecture [51]) enters the "eigen CNN" input, in which the non-linear transformation of the original data is realized without performing any multiplication operations. This improves the representativeness of these data, which is typical of every CNN. The output layers of "eigen CNN" are connected to the CCA block, where it is processed together, so that a high correlation between them is achieved. The CCA blocks have feedback to the layers of each "eigen CNN", which is obviously used in the learning process.

The architecture of "Deep Canonical Correlation Analysis" and "Deep generalized canonical correlation analysis" is shown in Fig. 10.

Thus, we can state that the idea of projection of digital images into their eigensubspaces reached the level of CNN architectures, including Deep NN.

CONCLUSION

The article presents the history of the appearance of projection methods into eigensubspaces (PMES), as a tool for processing experimental data and a tool for identifying and/or setig relationships in these data. PCA method based on the ideas of "eigenfaces" is described in detail, and through it – the methods of LDA, PLS and CCA are presented. The features and problems of the application of PMESs in the application to image processing of individuals are shown. As framework of the analysis of publications, the stages of the development of new approaches to PMES, which led to the appearance of ideas and methods of two-dimensional projection, oriented to image processing, as two-dimensional objects. In the technical literature, these methods are presented as variations from the abbreviations 2DPCA, 2DLDA, 2DPLS and 2DCCA. Algorithms of practical implementation of two-dimensional PMES are considered and four main branches of their development are revealed: two branches of iterative algorithms – in two independent variants and two branches of non-iterative algorithms – based on the direct solution of eigenvalue problems and it solution using SVD procedures. Their advantages and disadvantages are evaluated as well as examples of their use in the practice of FI recognition. It is also noted that the idea of projection of digital images into their eigensubspaces came to the level of CNN architectures, including Deep NN. Examples of corresponding solutions are shown and their brief analysis is given.

Particular attention is paid to the group of methods in which the two-dimensional projection into eigenspaces is defined as a two-stage procedure: 2DPCA/2DKLT, 2DLDA/2DKLT, 2DPLS/2DKLT and 2DCCA/2DKLT. At the first stage, **the analysis** of the original data is realized, including: the formation of common scattering matrices in accordance with a given criterion; solving eigenvalue problems with the calculation of projection matrices; definition of principal components. At the second stage, the full (or truncated-only by principal components) **two-dimen-**

sional Karunen-Loeve transformation is realized. Such a record more accurately reflects all the processes realized within the framework of two-dimensional projection methods, without mixing them and without substituting some processes and concepts by others (which can often be found in the technical literature). And the same record marks out a branch of two-dimensional projection methods with direct (not iterative) solution of eigenvalue problems from three other branches. Finally, such a record makes it easy to imagine (compose, implement) a generalized algorithm for the problem of two-dimensional PMES. In this case, canonical covariance matrix of the original data is formed, the blocks of which are used to construct scattering matrices in individual methods, and in 2DKLT procedure only projection matrices are changed, since its implementation is common to all methods.

Further research will be related to the implementation and use of a generalized algorithm for the problem of two-dimensional PMES.

REFERENCES

1. K. Pearson, "On lines and planes of closest fit to systems of points in space," London Edinburgh Dublin Philos. Mag. J. Sci., 6th ser. **2** (11), 559–572 (1901).
2. H. Hoteling, "Analysis of complex variables into principal components," J. Edu. Psychol. No. **24**, 417–441 (1933).
3. R. A. Fisher, "The use of multiple measurements in taxonomic problems," Ann. Eugenics **7** (2), 179–188 (1936).
4. H. Hoteling, "Relations between two sets of variates," Biometryka **28**, 321–377 (1936).
5. L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," J. Optical Soc. America, Ser. A (Optics, Image Science and Vision) **4** (3), 519–524 (1987).
6. M. Turk and A. Pentland, "Eigenfaces for recognition," J. Cogn. Neurosci. **3** (1), 71–86 (1991).
7. P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," IEEE Trans. Pattern Anal. Mach. Intellig. **19** (7), 711–720 (1997).
8. N. Tsapatsoulis, V. Alexopoulos, and S. Kollias, "A vector based approximation of KLT and its application to face recognition," in *Proc. 9th European Signal Processing Conf.* (*EUSIPCO 1998*) (Rhodes Palace, Island of Rhodes, Greece, Sept. 8-11, 1998), Vol. III, pp. 1581–1584.
9. G. A. Kukharev, *Biometric Systems: Methods and Means of People Identification* (Politekhnica, St. Petersburg, 2001) [in Russian].
10. G. Kukharev and A. Kuźmiński, *Biometric Techniques, Part I: Face Recognition Methods* (Wydział Informatyki, Politechnika Szczecińska, Szczecin, 2003) [in Polish].
11. G. Kukharev and P. Forczmański, "Data dimensionality reduction for face recognition," Machine Graphics and Vision **13** (1/2), 99–121 (2004).
12. G. Kukharev and P. Forczmański, "Face recognition by means of two-dimensional direct Linear Discriminant Analysis," in *Proc. 8th Int. Conf. Pattern Recognition and Information Processing* (*PRIP'2005*) (Minsk, May 18–20, 2005), pp. 280–283.
13. G. Kukharev and P. Forczmański, "Two-dimensional LDA approach to image compression and recognition," Computing, Multimedia & Intelligent Techniques, **2** (1), 45–53 (2006).
14. G. A. Kukharev and N. L. Shchegoleva, Human Face Recognition Systems (St. Petersburg Electrotechnical University "LETI", St. Petersburg, 2006) [in Russian].
15. G. Kukharev, A. Tujaka, and N. Binh, "System of face recognition using LDA with one training image per person," Metody Informatyki Stosowanej **16** (3), 167–185 (2008).
16. N. L. Shchegoleva and G. A. Kukharev, "Application of two-dimensional principal component analysis for recognition of face images," Pattern Recogn. Image Anal. **20** (4), 513–527 (2010).
17. J. Yang, D. Zhang, A. F. Frangi, and J. Yang, "Two-dimensional PCA: a new approach to appearance-based face representation and recognition," IEEE Trans. Pattern Anal. Mach. Intellig. **26** (1), 131–137 (2004).
18. D. Zhang and Z.-H. Zhou, "$(2D)^2$PCA: Two-directional two-dimensional PCA for efficient face representation and recognition," Neurocomputing **69**, 224–231 (2005).
19. J. Ye, "Generalized low rank approximations of matrices," Mach. Learning **61** (1), 167–191 (2005).
20. H. Kong, L. Wang, E. K. Teoh, et al., "Generalized 2D principal component analysis for face image representation and recognition," Neural Networks **18**, 585–594 (2005).
21. C. Ding and J. Ye, "2-Dimensional singular value decomposition for 2D maps and images," in *Proc. 2005 SIAM Int. Conf. on Data Mining* (Newport Beach, CA, April 21–23, 2005), pp. 32–43.
22. Z. Gu, W. Lin, et al., "Two-dimensional Singular Value Decomposition (2D-SVD) based video coding," in *Proc. 2010 17th IEEE Int. Conf. on Image Processing* (*ICIP*), (Hong Kong, Sept. 26–29, 2010), pp. 181–184.
23. K. S. Gurumoorthy, A. Rajwade, A. Banerjee, and A. Rangarajan, "A method for compact image representation using sparse matrix and tensor projections onto exemplar orthonormal bases," IEEE Trans. Image Process. **19** (2), 322–334 (2010).
24. K. Inoue and K. Urahama, "Equivalence of non-iterative algorithms for simultaneous low rank approximations of matrices," in *Proc. 2006 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (*CVPR'06*) (New York, NJ, June 17–22, 2006), Vol. 1, pp. 154–159.
25. X. Tang and X. Wang, "Face sketch recognition," IEEE Trans. Circuits Syst. Video Technol. **14** (1), 50–57 (2004).

26. CUFS dataset [Electronic resource]. http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html (accessed: 25.04.2015).

27. G. Kukharev and A. Oleinik, "Face photo-sketch transformation and population generation," in *Computer Vision and Graphics, Proc. Int. Conf. ICCVG 2016*, Ed. by A. Datta, R. Kozera, and K. Wojciechowski, *Lecture Notes in Computer Science* (Springer, Cham, 2016), Vol. 9972, pp. 329–340.

28. M. Borga, *Learning Multidimensional Signal Processing*, PhD Thesis, Linköping Studies in Science and Technology, Dissertation No. 531 (Linköping University, Linköping, Sweden, 1998).

29. M. Reiter, R. Donner, G. Langs, and H. Bischof, "3D and infrared face reconstruction from RGB data using Canonical Correlation Analysis," in *Proc. 18th Int. Conf. on Pattern Recognition* (*ICPR 2006*) (Hong Kong, August 20–24, 2006), Vol. 1, pp. 425–428.

30. M. Reiter, R. Donner, G. Langs, and H. Bischof, "Estimation of face depth maps from color textures using Canonical Correlation Analysis," in *Proc. Computer Vision Winter Workshop 2006* (Telč, Czech Republic, February 6–8, 2006), Ed. by O. Chum and V. Franc (Czech Pattern Recognition Society, 2006), 5 pages.

31. R. Donner, M. Reiter, G. Langs, et al., "Fast active appearance model search using canonical correlation analysis," IEEE Trans. Pattern Anal. Mach. Intellig. **28** (10), 1960–1964 (2006).

32. J. A. Ybáñez Zepeda, F. Davoine, and M. Charbit, "Face tracking using canonical correlation analysis," in *Proc. 2nd Int. Conf. on Computer Vision Theory and Applications* (*VISAPP 2007*) (Barcelona, Spain, March 8–11, 2007), Vol. 2, pp. 396–402.

33. A. Sharma and D. W. Jacobs, "Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch," in *Proc. 24th IEEE Conf. on Computer Vision and Pattern Recognition* (*CVPR 2011*) (Colorado Springs, CO, June 20–25, 2011), pp. 593–600.

34. S. H. Lee and S. Choi, "Two-dimensional Canonical Correlation Analysis," IEEE Signal Process. Lett. **14** (10), 735–738 (2007).

35. C. Zou, N. Sun, Z. Ji, and L. Zhao, "2DCCA: A novel method for small sample size face recognition," in *IEEE Workshop on Application of Computer Vision* (*WACV'07*), (Austin, TX, February 21–22, 2007), p. 43.

36. G. Kukharev and E. Kamenskaya, "Two-dimensional canonical correlation analysis for face image processing and recognition," Metody Informatyki Stosowanej **20** (3), 103–112 (2009).

37. G. Kukharev, A.Tujaka, and P. Forczmański, "Facial portraits matching by means of two-dimensional CCA and PLS," Metody Informatyki Stosowanej **22** (1), 69–86 (2010).

38. G. Kukharev and E. Kamenskaya, "Application of two-dimensional canonical correlation analysis for face image processing and recognition," Pattern Recogn. Image Anal. **20** (2), 210 –219 (2010).

39. P. Forczmański, G. Kukharev, and E. Kamenskaya, "Application of cascading two-dimensional canonical correlation analysis to image matching," Control Cybern. **40** (3), 833–848 (2011).

40. G. Kukharev, A. Tujaka, and P. Forczmański, "Face recognition using two-dimensional CCA and PLS," Int. J. Biometr. **3** (4), 300 –321 (2011).

41. G. A. Kukharev, E. I. Kamenskaya, Yu. N. Matveev, and N. L. Shchegoleva, *Methods of Facial Images Processing and Recognition in Biometrics*, Ed. by M. V. Hitrov (Politekhnica, St. Petersburg, 2013) [in Russian].

42. G. A. Kukharev, N. L. Shchegoleva, and E. I. Kamenskaya, "Representation and comparison methods for semantically different images," Pattern Recogn. Image Anal. **24** (4), 518–529 (2004).

43. Dogs and People. http://www.house-dog.ru/about_391.html [in Russian]

44. L. An and B. Bhanu, "Face image super-resolution using 2D CCA," Signal Process. **103**, 184–194 (2014).

45. S. Hou, Q. Sun, and D. Xia, "A two-dimensional Partial Least Squares with application to biological image recognition," in Proc. 6th Int. Conf. on Natural Computation (ICNC 2010) (Yantai, China, August 10–12, 2010), Vol. 1, pp. 57–61.

46. J. Qiu, H. Wang, et al., "Neural network implementations for PCA and its extensions," ISRN Artif. Intell. **2012**, Article ID 847305, 19 pages (2012). DOI: doi 10.5402/2012/847305

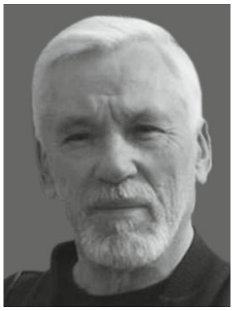
47. T.-H. Chan, K. Jia, S. Gao, et al., "PCANet: A simple deep learning baseline for image classification?" in IEEE Trans. Image Process. **24** (12), 5017–5032 (2015). arXiv:1404.3606 [cs.CV]

48. L. Tian, C. Fan, and Y. Ming, "Multiple scales combined principle component analysis deep learning network for face recognition," J. Electron. Imag. **25** (2), 023025 (16 pages) (2016). doi: 10.1117/1.JEI.25.2.023025

49. R. Hasegawa and K. Hotta, "PLSNet: A simple network using Partial Least Squares regression for image classification," in *Proc. 2016 23rd Int. Conf. on Pattern Recognition* (*ICPR*) (Cancún, Mexico, December 4-8, 2016), pp. 1601–1606.

50. G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep Canonical Correlation Analysis," in *Proc. 30th Int. Conf. on Machine Learning* (Atlanta, Georgia, USA, June 17–19, 2013), Proceedings of Machine Learning Research (PLMR) **28** (3), 1247–1255 (2013).

51. A. Benton, H. Khayrallah, B. Gujral, et al., "Deep Generalized Canonical Correlation Analysis," submitted to *5th Int. Conf. on Learning Representations* (*ICLR 2017*), 14 pages. arXiv:1702.02519 [cs.LG]



Georgy A. Kukharev Born in Leningrad, Russia. Received Ph.D. degree (1997) from the Fine Mechanics and Optics Institute (Leningrad, Russia) and Doctor of Technical Science degree (1986) from the Institute of Automatics and Computer Facilities (ABT, Riga, Latvia), Full Professor (2006). From 1993 he works at Szczecin University of Technology, Faculty of Computer Science and Information Systems (Poland). From 2003 he works at Saint Petersburg State Electrotechnical University LETI, Department of Computer Software Environment. In 2001–2003 visiting professor of Ecole Centrale de Lyon, Department of Mathematics and Computer Science. Since 2005 visiting professor of Hanoi University of Technology, Department of International Training Programmer. Author of more ten monographs, over 100 scientific papers, and over 44 patents in the areas: computer architecture of signal processing, image processing, and pattern recognition. Current interests: biometrics, including face detection and face recognition, Face/Sketch Retrieval, facial bar-code.

Nadegda L. Shchegoleva was born in Komsomolsk-on-Amur, Russia. Doctor of Science. From 2001–2006 she was Senior Researcher in Federal State Unitary Enterprise Central Research Institute "Morphyspribor" (JSC "Concern" "OKEANPRIBOR"). From 2007 she is Associate Professor in the Department of Software Engineering and Compute Application (Saint Petersburg Electrotechnical University "LETI", St.-Petersburg, Russia. Co-author of two monographs, 4 inventions, more than 60 scientific articles. Her research interests include biometric identification systems and access control systems, face recognition, synthesis and modeling of the recognition systems, Face/Sketch Retrieval, facial bar-code.