MINISTRY OF EDUCATION
OVIDIUS UNIVERSITY OF CONSTANTA
DOCTORAL SCHOOL OF APPLIED SCIENCES

PhD THESIS
SUMMARY

The study of some algebraic reconstruction techniques with applications in image reconstruction and image processing

Scientific Coordinator
Prof. univ. dr. Constantin Popa

Phd Student
Lăcrămioara (Liţă) Grecu

Constanta
2013

Introduction

In recent years due to technical evolution, problems related to image reconstruction and image processing (pattern/face recognition) have gained great interest in a wide range of applications. Image reconstruction problems occur in areas such as medicine (computed tomography), geology (electromagnetic geotomography), etc.. Face recognition is a matter of great interest in areas such as security and surveillance issues, forensic (criminal) problems, human-computer interface, multimedia communications, and so on.

This thesis treats the two topical directions:

1. image reconstruction, namely the demonstration, by a way of proof completely different from the one in [28], of the convergence of **Simultaneous Algebraic Reconstruction Technique** (SART) algorithm by integrating this algorithm in a class of general projective methods;

2. face recognition, more precisely we propose two new algorithms to solve the problem of face recognition. The first algorithm is a Principal Component Analisys (PCA) like algorithm (based on projecting the database onto a subspace of smaller dimension) and is a customization of the algorithm presented in [6] and [7]. The second algorithm is based on multilinear analysis and it is suitable for large datasets, where the number of pictures in the dataset is greater than the resolution of an image.

The thesis is structured into five chapters as follows.

In the *first chapter* are presented notions and concepts based on linear algebra and numerical analysis used throughout the thesis. We present notions about vector and matrix norms, singular value decomposition, orthogonal and oblique projections, least squares problems. It is also presented the problem of image algebraic reconstruction.

In this first chapter is also presented an algorithm for the construction of the scanning matrix in Electromagnetic Geotomography (EGT). We describe the algorithm for the construction of the scanning matrix for different types of scanning: "well to well", "surface to well" and total scanning. For all three types of scanning we present results regarding the number of nonzero elements of the scanning matrix. Moreover, we describe an optimization of the construction algorithm, which will reduce the running time of the algorithm.

In the *second chapter* are introduced the image Algebraic Reconstruction Techniques (ART), methods based on successive projections and simultaneous projections. Furthermore, we study the convergence of the algebraic reconstruction techniques based on successive projections.

We also study the convergence of some extensions of the algebraic reconstruction techniques to inconsistent problems and constrained problems. This

study is being done through a general projective method, presented in [38], which includes the known algebraic reconstruction methods: Kaczmarz, Cimmino, etc..

Chapter three deals with the study of the convergence of the algebraic reconstruction techniques based on simultaneous oblique projections, more precisely, the study of the convergence of SART algorithm. SART algorithm was introduced by A. H. Andersen and A. C. Kak in 1984 (see [1]) as a simultaneous alternative to Kaczmarz algorithm which is based on successive orthogonal projections.

The SART algorithm is based on simultaneous oblique projections and error correction terms are applied simultaneously for all rays of a particular projection.

The proof of convergence is made by integrating the SART algorithm to the general projective method presented in the second chapter, a new way of proof. Integrating the SART algorithm to the general projective method allows us to consider the SART algorithm with constraints, which will give us better reconstructions.

In *chapter four* are presented classes of algorithms for pattern recognition. We describe algorithms based on projection onto a subspace of smaller dimension. At the beginning of the chapter we present the eigenfaces algorithm which is the the oldest method (see [26], [29] and [40]) used for face recognition, but still has good results.

PCA (Principal Component Analisys) is a mathematic process that allows us to obtain from an initial large data set, another smaller data set with similar properties. This new data are called principal components. These components are chosen in descending order of their importance: the first component has the highest significance, the second component corresponds to the next eigenvalue (in magnitude) and so on. At the same time it is taken into account the restriction that each principal component is orthogonal to all previous principal components.

Next, we propose a new algorithm for face recognition, COD-A1 algorithm. This algorithm is a customized version of the algorithm descrided in [6] and [7]. Starting from the idea of low rank approximation from [6] and [7], and the idea of representing all images in a subspace with smaller dimensions, we propose the COD-A1 algorithm, which as can be seen in the experiments chapter, has a higher recognition rate than PCA algorithm.

Last but not least we propose a new algorithm based on tensors, algorithm A2. This algorithm is more suitable for datasets where the set of images is well organized and the number of photos in the dataset is greater than the resolution of an image, which has not been considered so far because the datasets available had small sizes (the number of images in the dataset was small). This algorithm based on tensors has a higher recognition rate than the

one of PCA algorithm.

The motivation why tensors are used in face recognition is that often the data is stored as a tensor. Instead of storing all the data as a matrix, we store all photos belonging to the same person as a matrix, and all matrices coresponding to all persons form a tensor. This leads to a better ordering of the pictures in the dataset.

Chapter five is reserved to numerical experiments. In this chapter are presented tests performed with the algorithm for the construction of the scanning matrix in Electromagnetic Geotomography from Chapter one, comparisons between the results obtained with SART algorithm (presented in Chapter three) and those obtained with Kaczmarz algorithm, standard ART class algorithm, for which the experiments are performed on two "ghosts" commonly used in the literature (see [24]) and experiments with all algorithms described in Chapter four, carried out on four databases, two own databases (CTOVF and CTOVD) and two databases known in the literature (ORL and ExtYaleB).

The original results presented in this thesis are included in the following papers:

- [16] Popa C., Grecu L., *Constrained SART algorithm for invers problems in image reconstruction*, Inverse Problems and Imaging, 1(7) (2013), 199-216. (Relative Influence Score = 2.13, Impact Factor = 1.074)

- [18] Grecu L., *On the construction of the scanning matrix in Electromagnetic Geotomography*, Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara, Seria Matematică - Fizică, 1 (2012), 42-56.

- [17] Grecu L., Nicola A., *Some results on simultaneous algebraic techniques in image reconstruction from projections*, ROMAI Journal, 2 (2009), 79-96, http://rj.romai.ro/

- [41] Pelican E., Grecu L., *Comparison Between Some Matrix Methods with Applications in Pattern Recognition*, presented at Applied Linear Algebra Conference, May 24-28, 2010, Novi Sad, Serbia.

- [20] Grecu (Liţă) L. , Pelican E., *Customized Orthogonalization via Deflation Algorithm with Applications in Face Recognition*, submitted to Carpathian Journal of Mathematics, 2013.

- [21] Grecu (Liţă) L. , Pelican E., *A Low-Rank Tensor-Based Algorithm for Face Recognition*, submitted to Pattern Recognition Letters, 2013.

- [43] Pelican E., Grecu (Liţă) L., *Solving the Pattern Recognition Problem with some Low-Rank Approximation Based Algorithms*, presented at XIème Colloque Franco-Roumain de Mathématiques Appliquées, 24-30 August, 2012, Bucureşti.

- [19] Grecu (Liţă) L. , Pelican E., *Systematic and comparative experiments with some algorithms for pattern recognition*, accepted for publication in Proceedings of the Nineth Workshop on Mathematical Modelling of Environmental and Life Sciences Problems, November 1-4, 2012, Constanţa, Romania.

- [42] Pelican E., Grecu L., *Low-Rank Matrix Methods in Pattern Recognition*, presented at Balkan Conference on Operational Research (BALCOR), 2009, published electronically with ISBN: 973-86979-9-9.

Constanţa, march 2013

Lăcrămioara (Liţă) Grecu

1 Preliminaries

1.1 Introductory notions

In this section are presented notions and concepts based on linear algebra and numerical analysis used throughout the thesis: vector and matrix norms, singular value decomposition (SVD), orthogonal and oblique projections in $\mathbb{R}^n$, linear least squares problems.

We will use the notations: A_i row i of matrix A, A^j column j of matrix A, A_{ij} the (i,j) entry of matrix A, I_n identity matrix of order n, A^T the transpose of matrix A, $\mathcal{N}(A) = \{x \in \mathbb{R}^n,\ Ax = 0\}$, $\mathcal{R}(A) =$
$\{y \in \mathbb{R}^m,\ \exists\ x \in \mathbb{R}^n,\ y = Ax\}$. The notation $S \oplus T$ is used for the orthogonal direct sum of the subspaces S and T, i.e. $E = S \oplus T \Leftrightarrow$

$$\begin{aligned} &(i)\ \forall\ x \in E,\ \exists\ s \in S,\ t \in T \text{ such that } x = s + t; \\ &(ii)\ S \cap T = \{0\}; \\ &(iii)\ \forall\ s \in S,\ t \in T,\ \langle s, t \rangle = 0. \end{aligned} \tag{1}$$

1.2 Image Algebraic Reconstruction

The mathematical formulation of the image algebraic reconstruction problem consists in finding a function of two variables when are known its projections, meaning its curvilinear integrals for any line of the plan. There are two forms: the direct problem and the inverse problem.

The direct problem involves calculating the integral of a (known) function f on a plane curve C, denoted by

$$I_C(f) = \oint_C f(x,y)\, dI. \tag{2}$$

For the inverse problem is assumed that we know the value of the integral $I_C(f)$ from (2), i.e. we can determin the values of the absorption function $f(x,y)$ in any point (x,y) from the domain.

Lambert's law states that there is a logarithmic dependence between the transmission intensity (I_S) of the ray and the reception one (I_R). In this case, for a ray SR we have

$$\ln\left(\frac{I_R}{I_S}\right) = \oint_{SR} f(x,y)\, dI. \tag{3}$$

To solve this problem of image algebraic reconstruction are considered two levels of discretization: discretization of the set of values of the absorption function f (discretization of the image) and discretization of lines SR in the plane (discretization of the scan). Discretization of the image involves dividing

the image in picture elements called pixels. Discretization of the scan involves a finite-ray scanning, $SR \in \mathbb{R}^2$.

If we use a total of m rays and a discretization of the image in n pixels, scanning matrix A is of size $m \times n$. Matrix A is sparse (see Lemma 1), doesn't have full rank, and is ill conditioned.

Measuring the intensity of the emission sources (S_i) and reception sources (in detectors R_j) for the i-th X-ray, we get the component b_i of the right hand side term b of the discrete model. This technique reduces image reconstruction to solving the least squares problem: determine $x \in \mathbb{R}^n$ such that.

$$\|Ax - b\| = \min \{\|Az - b\|, \ z \in \mathbb{R}^n\}, \tag{4}$$

or briefly

$$\|Ax - b\| = \min! \tag{5}$$

is called **algebraic reconstruction**. To solve (5) it has been developed a class of iterative methods based on projections, called **Algebraic Reconstruction Techniques (ART)**.

1.3 On the construction of the scanning matrix in Electromagnetic Geotomography

The results in this section were published in [18] L. Grecu (Liţă), *On the construction of the scanning matrix in Electromagnetic Geotomography*, where we present an algorithm for the construction of the scanning matrix. The scanning procedure was developed particulary in the field of Computed Tomography (see [24]). However, the idea was taken up successfully also in other areas such as Electromagnetic Geotomography (EGT).

In Electromagnetic Geotomography the scanning procedure consists in drilling two parallel wells, and scanning the cross section between them, using transmitters and receivers placed along the two wells. This method gives us information about cracks appeared after earthquakes or cleaves of underground pipelines. After scanning the desired section, a scanning matrix is obtained. This matrix can be used in various image reconstruction algorithms, which gives us information about the unknown structure of the underground (see [11]).

The geotomographic scanning process involves a set of transmitters $(S_1, \ S_2, \ \ldots, \ S_p)$ and a set of receivers $(R_1, \ R_2, \ \ldots, \ R_q)$ located in two parallel drilling wells. From each transmitter a scanning ray is send to every receiver. This is called well to well scanning.

The discrete model of the algebraic reconstruction problem will be a linear least squares problem of the form (4) (see [11], [24]), where A is the scanning matrix, and each element A_{ij} is obtained by measuring the length of the segment determined by the intersection of the i-th ray with the j-th pixel.

Measuring the intensities of the source emissions ($S_1,\ S_2,\ \ldots,\ S_p$) and reception (in the receivers $R_1,\ R_2,\ \ldots,\ R_q$) of the i-th electromagnetic ray, we will obtain the component b_i of the right hand side b of the discret model (4). We denote by x_{LS} the minimum norm solution of the least squares problem (4), given by $x_{LS} = A^+ b$, where A^+ is the Moore−Penrose pseudoinverse of matrix A (see e.g.[15]).

An image of dimension $\alpha \times \alpha$ is divided into pixels numbered row-wise $P_1,\ P_2,\ \ldots,\ P_n$. To this image one associates a matrix A, called scanning matrix, of size $m \times n$, where m is the number of transmitters multiplied by the number of receivers, and n is the number of pixels. The matrix A is constructed row by row and every row corresponds to a scanning ray. Moreover we will suppose that the pixels are normalized, i.e. with the edge length equal to 1.

1.3.1 The well to well procedure

For this method of obtaining the scanning matrix we have assumed that the number of transmitters and the number of receivers are equal, and that the transmitters/receivers are positioned in the middle of the pixel's boundary.

From each transmitter a scanning ray is send to every receiver. Let $P_{j_1},\ \ldots,\ P_{j_q}$ be the pixels that intersect the scanning ray i. The equation of the scanning ray i is obtained from the coordinates of the points which correspond to the transmitter and the receiver of this ray. Then, with the help of this equation of the scanning ray, it is calculated the length of the segment determined by the intersection of the i-th ray with the pixel P_j.

The i-th row of matrix A corresponds to the i-th ray. If the i-th ray intersects the j-th pixel, then the element A_{ij} will be equal with the length of the intersection between the i-th ray and the j-th pixel; if the i-th ray does not intersect the j-th pixel, then $A_{ij} = 0$. As we will show in Lemma 1, the matrix A is sparse.

Remark 1 *The same procedure can also be applied to a rectangular image of size $\alpha \times \beta$. In such a case the number of pixels in image is $n = \alpha \times \beta$, and the method for generating the scanning matrix is the same as for a square image.*

The next Lemma (given without proof in [32]) shows that the scanning matrix A is sparse.

Lemma 1 *[32] For the well to well case, the scanning matrix A has at most $m + n - 1$ nonzero elements on a row, for an image with size $n \times m$.*

1.3.2 The surface to well procedure

For better results with the reconstruction algorithms, we can also use surface trasmitters. In this case the receivers are positioned on both drilling wells and

the rays are sent from surface to both sides. This scanning is called surface to well scanning. Using this scanning procedure we get additional informations on the underground in comparison to well to well scanning.

Lemma 2 *The scanning matrix, A, associated to a surface to well scanning of an image of size $n \times m$, has at most $m + n - 1$ nonzero entries on a row.*

1.3.3 Total scanning

The total scanning procedure includes well to well scanning as well as surface to well scanning. This scanning procedure gives the most satisfactory results.

For the total scanning procedure remains valid rule that the i-th row of matrix A coresponds to the i-th ray. In this case the ray i can be a well to well ray or a surface to well ray. The density structure of the corresponding scanning matrix is given by the following result which holds directly from Lemmas 1 and 2.

Lemma 3 *For the total scanning procedure, the obtained scanning matrix A has at most $m+n-1$ nonzero elements on a row, for an image with size $n \times m$.*

1.3.4 Optimization of the construction algorithm

In this section we will present a procedure for the optimization of the algorithm for the construction of the scanning matrice. When constructing this matrix, we must take into account the area of the image which the ray is most likely to cross and focus on the pixels in that area. There is no need to take one by one each pixel of the image, and see if the ray intersects it. In this way we minimize the computational time.

To find the area that the ray is most likely to cross pixels we must focus on the rows of the image, where the transmitter and the receiver are positioned, and search all the pixels from those rows and from the rows between them. This can be done by taking into account the number of transmitters/receivers from a pixel (if there are more than one transmitter/receiver in a pixel they are positioned equidistant). Because the transmitters/receivers are placed in a specific order and for each scanning ray we know the corresponding transmitter and receiver, we can deduct the image rows where the transmitter/receiver is positioned.

The algorithm for finding the strip where the ray crosses the image is the following (see Figure 1, $\alpha \times \alpha$ is the image resolution and we suppose that $\alpha_i > \alpha_j$):

Step 1. Find the interval i on the edge AD where the source $S_i(0,\alpha_i)$ is positioned, $i = \alpha - floor(\alpha_i)$ with $\alpha_i \in (0, \alpha)$;
Step 2. Find the interval j on the edge BC where the receptor $R_j(\alpha,\alpha_j)$ is positioned, $j = \alpha - floor(\alpha_j)$ with $\alpha_j \in (0, \alpha)$;

Step 3. Obtain the strip $[i, j]$*;*
Step 4. The pixels in the strip are those with indices $(i-1)*\alpha+1, \cdots, j*\alpha$.

where $floor(\alpha)$ denotes the integer part of α.
Then we will analyse the intersection of the ray SR only with those pixels.

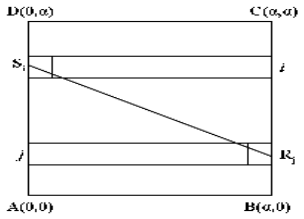

Figura 1: Optimization of the construction algorithm

For the surface to well procedure it is easier to determin the image area where the ray is more likely to cross. The number of pixels that are likely to be intersected by the ray decreases with 1, with each row. Hence we obtain o triangle of pixels, which may intersect the ray. The algorithm for determining those pixels is the following (we have $S(\alpha, i)$ and $R(0, j)$):

For each image row, $r = 1,\ 2, \ldots,\ j$*;*
the pixels of interest are: $(r-1)\,\alpha + 1, \ldots, (r-1)\,\alpha + j - (r-1)$

2 Solving the image algebraic reconstruction problem with iterative methods based on orthogonal projections

2.1 Kaczmarz and Cimmino Methods

Kaczmarz method appears first in 1937 in polish mathematician S. Kaczmarz paper [30], but a rigorous theoretical study of its properties is only made in 1971 by japanese mathematician K. Tanabe (see [50]). Kaczmarz method is also known as projections method because of its particular modality of construction of approximations $x^1,\ x^2,\ \ldots$.
We define $f_i(b;\cdot) : I\!R^n \to I\!R^n,\ i = \overline{1,m}$ and $F(b;\cdot) : I\!R^n \to I\!R^n$ by

$$f_i(b;x) = x - \frac{\langle x, A_i\rangle - b_i}{\|A_i\|^2} A_i = P_{H_i}(x), \tag{6}$$

and

$$F(b;x) = (f_1 \circ f_2 \circ \ldots \circ f_m)\,(b;x)\,. \tag{7}$$

Kaczmarz method is written as

$$x^{k+1} = F(b;x^k),\ k \geq 0, \tag{8}$$

where $x^0 \in I\!R^n$ is the initial aproximation.

Remark 2 *One iteration (8) of Kaczmarz method consists of successive (orthogonal) projections onto hyperplanes $H_m,\ H_{m-1}, \ldots, H_1$ associated with the equations of problem (4), in an a priori fixed order.*

The standard method for Algebraic Reconstructions Technique (ART) based on simultaneous projections is the algorithm proposed by G. Cimmino in [5]. The author considers instead of Kaczmarz's method orthogonal projections, the symmetrics of an approximation $x^k \in I\!R^n$ againts hyperplanes

$$H_i = \{x \in I\!R^n,\ \langle x, A_i \rangle = b_i\}\,.$$

We have

$$S_i(x) = x - 2\frac{\langle x, A_i \rangle - b_i}{\|A_i\|^2} A_i. \tag{9}$$

Cimmino method can be written as

$$x^{k+1} = \sum_{i=1}^{m} \frac{\omega_i}{\omega} S_i(x^k) = x^k - \frac{2}{\omega} \sum_{i=1}^{m} \omega_i \frac{\langle x^k, A_i \rangle - b_i}{\|A_i\|^2} A_i, \tag{10}$$

where $\omega_i > 0$ şi $\omega = \sum_{i=1}^{m} \omega_i$.

Hence x^{k+1} is a convex combination of the symmetrics $\left\{S_i(x^k),\ i = 1, \ldots, m\right\}$.

We will analyze convergence of Kaczmarz (8) and Cimmino (10) methods through a general approach presented in [38]. The authors consider there a general algorithm of the form: $x^0 \in I\!R^n$, for $k = 0, 1, \ldots$ we define the sequence

$$x^{k+1} = Tx^k + Rb, \tag{11}$$

where $T : n \times n,\ R : n \times m$, for matrix T and R are imposed the following properties

$$I - T = RA, \tag{12}$$

$$\forall\ y \in I\!R^m,\ Ry \in \mathcal{R}\left(A^T\right), \tag{13}$$

$$\text{if } \widetilde{T} = TP_{\mathcal{R}(A^T)} \text{ then } \left\|\widetilde{T}\right\| < 1. \tag{14}$$

Remark 3 *From (14) we get $T = TP_{\mathcal{R}(A^T)} + TP_{\mathcal{N}(A)} = \widetilde{T} + TP_{\mathcal{N}(A)}$, $P_{\mathcal{N}(A)}\widetilde{T} = P_{\mathcal{N}(A)}TP_{\mathcal{R}(A^T)} = 0$, şi $\widetilde{T}P_{\mathcal{N}(A)} = TP_{\mathcal{R}(A^T)}P_{\mathcal{N}(A)} = 0$.*

Proposition 1 *Kaczmarz method (6)-(8) can be written as (11), with* $T = P_1 P_2 \dots P_m$, *where* $P_i(x) = x - \frac{\langle x, A_i \rangle}{\|A_i\|^2} A_i$, *and* $R^i = \frac{1}{\|A_i\|^2} P_1 P_2 \dots P_{i-1}(A_i)$, *where* R^i *is the* i*-th column of matrix* R.

Cimmino method (10) can be written as (11), with $T = \sum_{i=1}^{m} \frac{\omega_1}{\omega} S_i$, *where* $S_i = I - 2\frac{A_i A_i^T}{\|A_i\|^2}$, *and* $R = \sum_{i=1}^{m} \frac{\omega_1}{\omega} \frac{1}{\|A_i\|^2} A_i$.

Proposition 2 *([38]) If (12)-(14) hold, then*
(i)

$$\text{If } x \in \mathcal{N}(A) \text{ then } Tx = x \in \mathcal{N}(A), \tag{15}$$

and

$$\text{if } x \in \mathcal{R}\left(A^T\right) \text{ then } Tx \in \mathcal{R}\left(A^T\right). \tag{16}$$

(ii) Matrix $I - \widetilde{T}$ *is invertible and matrix* $G : n \times m$ *defined by*

$$G = \left(I - \widetilde{T}\right)^{-1} R, \tag{17}$$

satisfies

$$AGA = A \tag{18}$$

and

$$GP_{\mathcal{R}(A)}(b) = x_{LS}. \tag{19}$$

(iii) Matrix T *has the properties*

$$\|Tx\| = \|x\| \Leftrightarrow x \in \mathcal{N}(A) \tag{20}$$

and

$$\|T\| \leq 1. \tag{21}$$

(iv) For the sequence $\left(x^k\right)_{k \geq 0}$ *given by (11) we have*

$$P_{\mathcal{N}(A)}(x^k) = P_{\mathcal{N}(A)}(x^0), \ \forall \ k \geq 0. \tag{22}$$

Theorem 1 *([38]) If (12)-(14) hold, the sequence* $(x^k)_{k \geq 0}$ *given by (11), with* $x^0 \in I\!R^n$ *converges, and*

$$\lim_{k \to \infty} x^k = P_{\mathcal{N}(A)}(x^0) + Gb, \tag{23}$$

where G *is given by (17).*

2.2 Extensions to constrained problems and inconsistent problems

In [32] is considered a function $C : I\!R^n \to I\!R^n$, with $Im\,(C) \subset I\!R^n$ closed and the following properties:

$$\|Cx - Cy\| \leq \|x - y\|\,, \tag{24}$$

$$\text{if } \|Cx - Cy\| = \|x - y\| \text{ then } Cx - Cy = x - y, \tag{25}$$

$$\text{if } y \in Im\,(C) \text{ then } Cy = y. \tag{26}$$

An example of this kind of function is the orthogonal projection onto a convex and closed set in $I\!R^n$.

Example 1 *$C : I\!R^n \to [a, b] = [a_1, b_1] \times \ldots \times [a_n, b_n]$, given by*

$$(Cx)_i = \begin{cases} a_i, & \text{if } x_i < a_i \\ x_i, & \text{if } x_i \in [a_i, b_i] \\ b_i, & \text{if } x_i > b_i \end{cases}. \tag{27}$$

Lemma 4 *We have $C = P_{[a,b]}$, hence satisfies (24)-(26).*

2.2.1 The general projection method with constraints

For the general projection method (11) we consider the constrained version:

$$x^0 \in I\!R^n,\ x^{k+1} = C\left(Tx^k + Rb\right),\ k \geq 0. \tag{28}$$

If ξ is the limit of the sequence $\left(x^k\right)_{k\geq 0}$ from (11) (see (23)), starting with $x^0 \in \mathcal{R}\left(A^T\right)$, and using (19) it results $\xi = Gb = GP_{\mathcal{R}(A)}\,(b) + GP_{\mathcal{N}(A^T)}\,(b) = x_{LS} + \delta$, where $\delta = P_{\mathcal{N}(A^T)}\,(b)$. We define the set $\mathcal{V} = \{y \in Im\,(C)\,,\ y - \delta \in LSS\,(A; b)\}$ and suppose that $\mathcal{V} \neq \emptyset$.

Remark 4 *If the problem (4) is consistent, then $\delta = 0$, and*

$$\mathcal{V} = S\,(A; b) \cap Im\,(C)\,. \tag{29}$$

The next lemma proves that the image of vector $h \in Im\,(C)$ by (28) is closer to $\mathcal{V}$ than the vector h.

Lemma 5 *[38] Let C with the properties (24)-(26). If $h \in Im\,(C)$ and*

$$g = C\,(Th + Rb)\,, \tag{30}$$

then, for any $y \in \mathcal{V}$

$$\|g - y\| \leq \|h - y\| \tag{31}$$

and, either

$$\|g - y\| < \|h - y\| \tag{32}$$

or

$$g = h \in \mathcal{V}. \tag{33}$$

Theorem 2 *[38] Let* C *with the properties (24)-(26). If* $x^0 \in Im\,(C)$ *and* $\left(x^k\right)_{k\geq 0}$ *is defined by (28), then* $\lim\limits_{k\to\infty} x^k$ *exists and belongs to* $\mathcal{V}$.

2.2.2 The extended general projection method

In [38] is considered the following extended version of the general projection method (11). Let matrices U and S of size $m \times m$, respectively $m \times n$, similar to matrices T and R from (11), for the system

$$A^T y = 0. \tag{34}$$

The iterative algorithm (11) becomes

$$y^0 \in I\!R^m,\ y^{k+1} = Uy^k + S \cdot 0 = Uy^k,\ \forall\ k \geq 0. \tag{35}$$

For algorithm (35) we have the following result of convergence (similar to (23)): for any $y^0 \in I\!R^m$, the sequence $(y^k)_{k\geq 0}$, given by (35), converges and

$$\lim_{k\to\infty} y^k = P_{\mathcal{N}(A^T)}(y^0). \tag{36}$$

If $y^0 = b$, from (36) we obtain $\lim\limits_{k\to\infty} y^k = P_{\mathcal{N}(A^T)}(b)$, hence $\lim\limits_{k\to\infty}(b - y^k) = P_{\mathcal{R}(A)}(b)$.

The extended general projection method

Let $x^0 \in I\!R^n,\ y^0 = b$; for $k = 0,\ 1,\ \ldots$ compute

$$y^{k+1} = Uy^k, \tag{37}$$

$$b^{k+1} = b - y^{k+1}, \tag{38}$$

$$x^{k+1} = Tx^k + Rb^{k+1}. \tag{39}$$

For algorithm (37)-(39), in [38] is proved the following convergence property.

Theorem 3 *[38] If matrices* T *and* R *satisfy properties (12)-(14),* $\forall\ x^0 \in I\!R^n$, *the sequence* $(x^k)_{k\geq 0}$ *given by (37)-(39) converges and*

$$\lim_{k\to\infty} x^k = P_{\mathcal{N}(A)}(x^0) + x_{LS} \in LSS(A; b). \tag{40}$$

2.2.3 The extended general projection method with constraints

In [38] is considered the following algorithm.

The extended general projection method with constraints

Let $x^0 \in Im(C),\ y^0 = b$; for $k = 0,\ 1,\ \ldots$ compute

$$y^{k+1} = Uy^k, \tag{41}$$

$$b^{k+1} = b - y^{k+1}, \tag{42}$$

$$x^{k+1} = C[Tx^k + Rb^{k+1}]. \tag{43}$$

If the set $\mathcal{V}^*$ defined by

$$\mathcal{V}^* = LSS(A; b) \cap Im(C) \tag{44}$$

is nonempty, it is proved that the sequence $\left(x^k\right)_{k\geq 0}$ given by (41)-(43) converges to an element from $\mathcal{V}^*$.

Lemma 6 *[38] For any $y \in \mathcal{V}^*$ we have the equalities*

$$(I - T)\, y = (I - T)\, x_{LS} = Rb_A, \tag{45}$$

with

$$b_A = P_{\mathcal{R}(A)}(b). \tag{46}$$

Lemma 7 *[38] Let $k \geq 0$ fixed and suppose that the application C satisfies (24) and (26). Then, for any $y \in \mathcal{V}^*$ we have*

$$\left\|x^{k+1} - y\right\| \leq \left\|T\left(x^k - y\right) - R\widetilde{U}^{k+1}b_A\right\|. \tag{47}$$

Moreover, the sequence $\left(x^k\right)_{k\geq 0}$ given by (41)-(43) is bounded.

From Lemma 7, we get that the sequence $\left(x^k\right)_{k\geq 0}$ given by (41)-(43) has a convergent subsequence $\left(x^{k_s}\right)_{s\geq 0}$, i.e.

$$\lim_{s\to\infty} x^{k_s} = u \in Im(C). \tag{48}$$

Lemma 8 *[38] In Lemma's 7 hypotheses, element u from (48) belongs to $\mathcal{V}^*$ from (44).*

Lemma 9 *[38] If the sequence $\left(x^k\right)_{k\geq 0}$ from (41)-(43) has a convergent subsequence $\left(x^{k_s}\right)_{s\geq 0}$, with the limit $u \in \mathcal{V}^*$, then any other convergent subsequence, $\left(x^{\bar{k}_s}\right)_{s\geq 0}$, converges to the same limit $u \in \mathcal{V}^*$.*

Theorem 4 *[38] If $\mathcal{V}^*$ from (44) is nonempty, (24) and (26) hold and matrix A has all rows and columns nonzero, then for any $x^0 \in \mathbb{R}^n$ the sequence $\left(x^k\right)_{k\geq 0}$ given by (41)-(43) converges to an element from $\mathcal{V}^*$.*

3 Convergence study of algebraic reconstruction techniques based on simultaneous oblique projections

In [16] L. Grecu, C. Popa, *Constrained SART algorithm for inverse problems in image reconstruction* is presented the inclusion of SART algorithm to the general projection method (11)-(14), which allows us to consider algorithm SART with constraints.

In [16] we obtain, though a completely different way, the convergence result from [27] and [28]. Integrating SART algorithm to the general projection method from [38] (presented in Chapter 2) we obtain not only its convergence, but furthermore, we can consider algorithm SART with constraints and prove its convegence.

SART algorithm was introduced by Andersen and Kak in 1984 in [1] as a simultaneous alternative to Kaczmarz algorithm, which is based on successive orthogonal projections. SART algorithm is based on simultaneous oblique projections. First proof of convergence of SART algorithm was given in [28].

SART algorithm: $x^0 \in I\!R^n$

$$x^{k+1} = x^k + \lambda_k V^{-1} A^T W \left(b - Ax^k\right), \ k \geq 0, \tag{49}$$

where V and W are positive definite diagonal matrices

$$V = diag\left(V_{11}, \ldots, V_{nn}\right), \ W = diag\left(W_{11}, \ldots, W_{mm}\right), \tag{50}$$

with

$$V_{jj} = \sum_{i=1}^{m} |A_{ij}|, \ j = 1, \ldots n, \ \frac{1}{W_{ii}} = \sum_{j=1}^{n} |A_{ij}|, \ i = 1, \ldots, m. \tag{51}$$

We will denote by $\langle \cdot, \cdot \rangle_V$, $\langle \cdot, \cdot \rangle_W$, $\|\cdot\|_V$, $\|\cdot\|_W$ the energy scalar products defined onto $I\!R^n$, $I\!R^m$ by matrices V and W and the corresponding norms. For the subspaces $S \subset I\!R^n$, $T \subset I\!R^m$ we will denote with P_S^V, P_T^W the orthogonal projections onto S, T with respect to $\langle \cdot, \cdot \rangle_V$, respectively $\langle \cdot, \cdot \rangle_W$, where $\langle x, y \rangle_V = \langle Vx, y \rangle$, $\|x\|_V = \sqrt{\langle x, x \rangle_V}$, $\|A\|_{V,W} = \sup_{\|x\|_V = 1} \|Ax\|_W$. The notation $S \oplus_{\perp_V} T$ will be used for the direct orthogonal sum of the subspaces S and T with respect to the energy scalar product $\langle \cdot, \cdot \rangle_V$, i.e. $E = S \oplus_{\perp_V} T \Leftrightarrow$

$$\begin{aligned} &(i) \ \forall \ x \in E, \ \exists \ s \in S, \ t \in T \text{ such that } x = s + t; \\ &(ii) \ S \cap T = \{0\}; \\ &(iii) \ \forall \ s \in S, \ t \in T, \ \langle s, t \rangle_V = 0. \end{aligned} \tag{52}$$

3.1 Convergences of SART algorithm

We will consider the matrix A and its adjoint A^τ as linear functions

$$A : (I\!R^n, \langle\cdot,\cdot\rangle_V) \to (I\!R^m, \langle\cdot,\cdot\rangle_W)\,,\ A^\tau : (I\!R^m, \langle\cdot,\cdot\rangle_W) \to (I\!R^n, \langle\cdot,\cdot\rangle_V) \tag{53}$$

with A^τ uniquely defined by the equality

$$\langle Ax, y\rangle_W = \langle x, A^\tau y\rangle_V\,,\ \forall\ x \in I\!R^n,\ y \in I\!R^m. \tag{54}$$

For SART algorithm (49) we consider the least squares problem: find $x \in I\!R^n$ such that

$$\|Ax - b\|_W = \min\left\{\|Az - b\|_W\,,\ z \in I\!R^n\right\}. \tag{55}$$

Let $LSS_{V,W}(A;b)$ the solutions set for (55), and $x_{LS}^{V,W}$ its minimum $\|\cdot\|_V$-norm solution. The problem

$$\left\|\frac{1}{\rho}Ax - \frac{1}{\rho}b\right\|_W = \min! \tag{56}$$

is equivalent to (55), and if ρ satisfies

$$\rho \le \min\left\{\frac{1}{\|A\|_{V,W}}, \frac{1}{\|A^\tau\|_{W,V}}\right\} \tag{57}$$

we get

$$\left\|\frac{1}{\rho}A\right\|_{V,W} \le 1. \tag{58}$$

Lemma 10 *The transpose A^τ of matrix A from (53) is given by*

$$A^\tau = V^{-1}A^TW. \tag{59}$$

Moreover, we have the decompositions

$$I\!R^n = \mathcal{N}(A) \oplus_{\perp_V} \mathcal{R}(A^\tau),\ I\!R^m = \mathcal{N}\left(A^\tau\right) \oplus_{\perp_W} \mathcal{R}\left(A\right), \tag{60}$$

where $\oplus_{\perp_V}$ and $\oplus_{\perp_W}$ are the direct orthogonal sums corresponding to the scalar products $\langle\cdot,\cdot\rangle_V$, and $\langle\cdot,\cdot\rangle_W$, respectively.

For

$$\lambda_k = \lambda,\ \forall\ k \ge 0, \tag{61}$$

we write the SART algorithm (49) as in (11), with $T : n \times n$ and $R : n \times m$ defined by

$$T = I - \lambda V^{-1}A^TWA,\ R = \lambda V^{-1}A^TW. \tag{62}$$

Lemma 11 *Matrices T and R from (62) have the following properties (similar with (12)-(14))*

$$I - T = RA, \tag{63}$$

$$\forall\ y \in I\!R^m,\ Ry \in \mathcal{R}\left(A^\tau\right), \tag{64}$$

$$\text{if } 0 < \lambda < 2 \text{ and } \widetilde{T} = T \cdot P^V_{\mathcal{R}(A^\tau)},\ \text{then } \left\|\widetilde{T}\right\|_{V,V} < 1, \tag{65}$$

where $\left\|\widetilde{T}\right\|_{V,V} = \sup_{\|x\|_V \leq 1} \left\|\widetilde{T}x\right\|_V$.

Proposition 3 *As in [38], but with respect to the decomposition (60) we get*

$$T = \widetilde{T} \oplus_{\perp_V} P^V_{\mathcal{N}(A)}. \tag{66}$$

Theorem 5 *If (63)-(65) hold then the following are true*
(i)

$$\text{If } x \in \mathcal{N}\left(A\right) \text{ then } Tx = x, \tag{67}$$

and

$$\text{if } x \in \mathcal{R}\left(A^\tau\right) \text{ then } Tx \in \mathcal{R}\left(A^\tau\right). \tag{68}$$

(ii) Matrix $I - \widetilde{T}$ is invertible and G defined by

$$G = \left(I - \widetilde{T}\right)^{-1} R, \tag{69}$$

satisfies

$$AGA = A \tag{70}$$

and

$$GP^W_{\mathcal{R}(A)}\left(b\right) = x^{V,W}_{LS}. \tag{71}$$

(iii) Matrix T has the properties

$$\|Tx\|_V = \|x\|_V \Leftrightarrow x \in \mathcal{N}\left(A\right). \tag{72}$$

(iv) For the sequence $\left(x^k\right)_{k\geq 0}$ given by (49) we have

$$P^V_{\mathcal{N}(A)}(x^k) = P^V_{\mathcal{N}(A)}(x^0),\ \forall\ k \geq 0. \tag{73}$$

(v) If (63)-(65) hold then $(x^k)_{k\geq 0}$ given by (49), with $x^0 \in I\!R^n$ converges, and

$$\lim_{k\to\infty} x^k = P^V_{\mathcal{N}(A)}(x^0) + x^{V,W}_{LS}, \tag{74}$$

where matrix G is given by (69).

Remark 5 *The simultaneously form of SART method allows complete parallelization of this algorithm. Considerations on this issue are presented in Chapter 5.*

3.2 SART algorithm with constraints

Including SART algorithm to the general projection method (11)-(14) allows us to consider SART algorithm with constraints (CSART). This result is not obtained in [27] or [28].

Let $C : I\!R^n \to I\!R^n$ an application (generally nonlinear) with $\mathrm{Im}\,(C) \subset I\!R^n$ closed. We shall replace the assumptions (24)-(26) with

$$\|Cx - Cy\|_V \leq \|x - y\|_V \,, \tag{75}$$

$$\text{if } \|Cx - Cy\|_V = \|x - y\|_V \text{ then } Cx - Cy = x - y, \tag{76}$$

$$\text{if } y \in \mathrm{Im}\,(C) \text{ then } Cy = y. \tag{77}$$

According to the paper [32] we define the algorithm SART with constraints (CSART).

CSART algorithm
Let $x^0 \in I\!R^n$, for $k = 0,\ 1,\ \ldots$

$$x^{k+1} = C\left(Tx^k + Rb\right). \tag{78}$$

From (23), for $x^0 \in \mathcal{R}\left(A^\tau\right)$ we have $\lim\limits_{k\to\infty} x^k = x_{LS}^{V,W}$, thus according to [32] we define the set $\mathcal{S} \subseteq I\!R^N$ by

$$\mathcal{S} = \mathrm{Im}\,(C) \cap LSS_{V,W}\,(A; b)\,, \tag{79}$$

and suppose that $\mathcal{S} \neq \emptyset$.

Theorem 6 *Let us suppose that the constraining function C satisfies (75)-(77). If $x^0 \in \mathrm{Im}\,(C)$ and $(x^n)_{n\geq 0}$ is given by (78), then $\lim\limits_{n\to\infty} x^n$ exists and belongs to $\mathcal{S}$.*

Proposition 4 *The application $C : I\!R^n \to [a, b] = [a_1, b_1] \times \ldots \times [a_n, b_n]$, defined by*

$$(Cx)_i = \begin{cases} a_i, & \text{if } x_i < a_i \\ x_i, & \text{if } x_i \in [a_i, b_i] \\ b_i, & \text{if } x_i > b_i \end{cases} .$$

is a constraining function and satisfies (75)-(77).

Remark 6 *The above constraining function is useful because in real applications we know the interval $[a_i, b_i]$ (see e.g. [24] for medical image reconstruction).*

Proposition 5 *The application* $C : I\!R^n \to S = [0, \infty) \times \ldots \times [0, \infty)$, *defined by*

$$(Cx)_i = \begin{cases} x_i, & \text{if } x_i \geq 0 \\ 0, & \text{if } x_i < 0 \end{cases} .$$

is a constraining function and satisfies (75)-(77).

4 Classes of algorithms for face recognition

In [20] L. Grecu (Liţă), E. Pelican, *Customized Orthogonalization via Deflation Algorithm with Applications in Face Recognition* and [21] L. Grecu (Liţă), E. Pelican, *A Low-Rank Tensor-Based Algorithm for Face Recognition* we address the pattern recognition problem.

For a better understanding of the algorithms described in this paper, we present first the problem we are addressing. For a dataset of images of P persons, we transform all images into vectors $\{\Gamma_1, \Gamma_2, \ldots, \Gamma_N\}$. These N images are split into two nonoverlapping subsets: training subset and testing subset. The problem under consideration is: given an image Γ of a person from the testing set, we want to see if the algorithm identifies that person using the pictures from the training set.

4.1 Eigenfaces algorithm (PCA)

Although this projection method is already well-known, we will include a brief description for the sake of completeness. The eigenfaces are the principal components of a set of faces, or equivalent eigenvectors of the covariance matrix of a set of faces (see [51] and [52]). These eigenvectors are called eigenfaces because when represented, they resemble human faces. A set of eigenfaces can be generated by performing a mathematical process called Principal Component Analysis (PCA), on a large set of images that represents different faces. These eigenvectors are chosen in descending order of their importance: the first component has the highest significance and so on. At the same time it is taken into account the restriction that each principal component is orthogonal to all previous principal components.

The principal components are given by the eigenvectors of the covariance matrix. The first principal component is the eigenvector corresponding to the largest eigenvalue, the second principal component is the eigenvector corresponding to the next largest eigenvalue and so on.

The idea of using principal components to represent human faces was developed by Sirovich and Kirby (see [31] and [49]) and used by Turk and Pentland (see [51] and [52]) for face detection and face recognition.

For a database with N images, every image has the same resolution $M = n_1 \times n_2$ and it is transformed into a vector Γ_i, whose size is $M \times 1$. Then we

compute the average face vector $\Psi = \frac{1}{N}\sum_{i=1}^{N}\Gamma_i$, and substract the average face vector from all vectors $\varphi_i = \Gamma_i - \Psi,\ i = 1, \ldots, N$ i.e. in statistical terminology, we "center" the data. As noted in [13] (page 116) the first singular image which is defined by the principal pair of singular vectors looks very much like the average vector. Thus, it seems that there is no gain in substracting the average vector. The same remark can be done for the COD-A1 algorithm. For the Eigenfaces algorithm we still follow the ideea from [51].

The PCA algorithm is applied to this set of large vectors $\varphi_1,\ \varphi_2, \ldots, \varphi_N$. The algorithm seeks a set of orthonormal vectors $u_1,\ u_2,\ \ldots,\ u_N$ which best describes the patterns that appear in the dataset.

Lemma 12 *[51] [53] The vector u_k is chosen so that the quantity*

$$\lambda_k = \frac{1}{N}\sum_{i=1}^{N}\left(u_k^T\varphi_i\right)^2 \tag{80}$$

is maximum having the restriction

$$u_i^T u_k = \delta_{ik} = \begin{cases} 1, & \text{if } i = k \\ 0, & \text{otherwise} \end{cases} \tag{81}$$

where the vectors u_k and the scalars λ_k are eigenvectors and eigenvalues of the covariance matrix $C = \frac{1}{N}AA^T$ for $A = [\varphi_1\ \varphi_2 \ldots \varphi_N]$.

The size of the covariance matrix $C = AA^T$ is $M \times M$, where M is the resolution of an image. Because in practice, the number M is very large, the computational effort to determine the M eigenvalues and M eigenvectors for matrix C is huge. In this case, the idea is to reduce the size and therefore the amount of calculations. Let $L = A^TA,\ L \in \mathbb{R}^{N\times N}$. Usually N, the number of images in the dataset is much smaller then the size of a vector, M, and in this case it is much easier to calculate N eigenvectors and N eigenvalues for a matrix of size $N \times N$.

For the matrix L we consider the eigenvectors v_i so $A^TAv_i = \mu_i v_i$. And obtain $AA^TAv_i = \mu_i Av_i$, so we get Av_i is an eigenvector for matrix C. We seek N eigenvectors, v_i, for matrix L. From all N vectors obtained only first K vectors corresponding to the largest K eigenvalues are sufficient to characterize the initial set of images. The other $N-K$ vectors corresponding to the smallest eigenvalues and the information associated with them is less significant so they are discarded. Consequently, the orthonormal vector base from $\mathbb{R}^M$ (which will be used to obtain the images/vectors): $u_1,\ u_2,\ \ldots,\ u_N$ will be truncated to $u_1,\ u_2,\ \ldots,\ u_K$, with $K << N$.

In order to choose the level of truncation K, we can use one of the three methods mentioned in [10]. In our experiments we use the first method, i.e. we discarded roughly the last 90% of the eigenvectors.

To identify a new image, Γ, we represent it using the eigenvectors $\{u_1,\ u_2,\ \ldots,\ u_K\}$. We have $\omega_i = u_i^T(\Gamma - \Psi),\ i = 1 : K$. These coefficients ω_i form the vector $\Omega^T = [\omega_1,\ \omega_2, \ldots, \omega_K]$. The vector Ω describes the contribution of each eigenface in representing the image Γ and is used to classify the new image Γ.

4.2 Nearest Neighbour (NN) algorithm

In this section we briefly present an algorithm used in pattern recognition. In Chapter 5 and in [19] Grecu (Liţă) L. , Pelican E., *Systematic and comparative experiments with some algorithms for pattern recognition* are presented comparisons between the results obtained with this algorithm and the ones obtained with the other algorithms presented in this chapter.

The simplest method that can be used in pattern recognition is sequential search, i.e. finding the nearest neighbour for the picture we are looking for. So let Γ be an image (picture of a person or a digit) we want to find the nearest picture from the database $\{\Gamma_1,\ \Gamma_2,\ \ldots,\ \Gamma_N\}$. Sequential search (Nearest Neighbor algorithm, see [36]) implies comparing them with each picture and find an index i_0 such that

$$\|\Gamma - \Gamma_{i_0}\| = \min_{1 \le i \le N} \|\Gamma - \Gamma_i\|. \tag{82}$$

4.3 COD-A1 algorithm

In [20] L. Grecu (Liţă), E. Pelican, *Customized Orthogonalization via Deflation Algorithm with Applications in Face Recognition* we propose a new algorithm for face recognition.

In order to improve (as recognition rate) the PCA algorithm (a truncated SVD algorithm), one has to focus on finding a good low-rank approximation for the matrix A. Even if the truncated SVD gives the best possible low-rank approximation in both the Frobenius norm and L2 norm, for large matrices, a full or thin SVD can be too expensive. An option is based on Lanczos' algorithm and it is suitable for handling large matrices. It is well-known that Lanczos' method has two potential drawbacks. First one, while it is expected to give good estimates of the largest and smallest eigenvalues (singular values), it has a difficulty to estimate the intermediate values. Second one, as the number of iterations increases, the computed singular vectors may lose orthogonality. This last issue is already solved by reorthogonalization. In [4] and [14] the authors have already used this Lanczos' method for such low rank approximation. But, as reported, the accuracy of their proposed methods does not improve the one of PCA.

Another option for computing a rank k truncated SVD is using orthogonalization via deflation algorithm proposed in [6] and [7]. For a matrix $A : M \times$

$N,\ M \geq N$, this method generates a sequence of matrices $A_1,\ A_2,\ \ldots, A_{k+1}$, for which

$$A_{k+1} = A_k - \tilde{\sigma}_k \tilde{u}_k \tilde{v}_k^T = A - \sum_{j=1}^{k} \tilde{\sigma}_j \tilde{u}_j \tilde{v}_j^T = A - \tilde{U}_k \tilde{D}_k \tilde{V}_k^T = A - \tilde{B}_k, \qquad (83)$$

where $\tilde{u}_k$, $\tilde{v}_k$ and $\tilde{\sigma}_k$ are given by (84), (85), and respectively (86), $\tilde{U}_k = [\tilde{u}_1\ \tilde{u}_2 \ldots\ \tilde{u}_k]$, $\tilde{V}_k = [\tilde{v}_1\ \tilde{v}_2 \ldots\ \tilde{v}_k]$, $\tilde{D}_k = diag\,(\tilde{\sigma}_1,\ \tilde{\sigma}_2, \ldots,\ \tilde{\sigma}_k)$, and $\tilde{B}_k = \tilde{U}_k \tilde{D}_k \tilde{V}_k^T$. Matrix $\tilde{B}_k$ serves as a low-rank approximation of matrix A.

Let $\hat{u}_k \in \mathcal{R}\,(A_k)$ and $\hat{v}_k \in \mathcal{R}\left(A_k^T\right)$ be an arbitrary pair of unit vectors that satisfy $\hat{u}_k^T A_k \hat{v}_k > 0$. We have

$$\tilde{u}_k = A_k \hat{v}_k / \left\| A_k \hat{v}_k \right\|_2, \qquad (84)$$

$$\tilde{v}_k = A_k^T \hat{u}_k / \left\| A_k^T \hat{u}_k \right\|_2, \qquad (85)$$

and

$$\tilde{\sigma}_k = \left(\left\| A_k \hat{v}_k \right\|_2 \left\| A_k^T \hat{u}_k \right\|_2 \right) / \left(\hat{u}_k^T A_k \hat{v}_k \right). \qquad (86)$$

Theorem 7 *(see [6] şi [7]) Let the matrices $\hat{U}_k \in I\!R^{M\times k}$, $\hat{V}_k \in I\!R^{N\times k}$, and $\hat{D}_k \in I\!R^{k\times k}$ be defined by the equalities*

$$\hat{U}_k = [\hat{u}_1\ \hat{u}_2\ \ldots\ \hat{u}_k],\ \hat{V}_k = [\hat{v}_1\ \hat{v}_2\ \ldots\ \hat{v}_k] \text{ and } \hat{D}_k = (\hat{\sigma}_1,\ \hat{\sigma}_2,\ \ldots,\ \hat{\sigma}_k), \qquad (87)$$

where $\hat{\sigma}_j = \hat{u}_j^T A \hat{v}_j$, for $j = 1,\ \ldots,\ r = rank(A)$. Then, in exact arithmetic, the following relations hold for $k = 1,\ \ldots,\ r$:

$$\mathcal{R}\,(A_{k+1}) \subseteq \mathcal{R}\,(A_k),\ \mathcal{R}\left(A_{k+1}^T\right) \subseteq \mathcal{R}\left(A_k^T\right), \qquad (88)$$

$$\mathcal{R}\left(\hat{U}_k\right) \subseteq \mathcal{R}\,(A),\ \mathcal{R}\left(\hat{V}_k\right) \subseteq \mathcal{R}\left(A^T\right), \qquad (89)$$

$$\hat{U}_k^T A_{k+1} = 0,\ A_{k+1} \hat{V}_k = 0, \qquad (90)$$

$$\hat{U}_k^T \hat{U}_k = I,\ \hat{V}_k^T \hat{V}_k = I. \qquad (91)$$

Thus, for $k = r$ the columns of $\hat{U}_r$ and $\hat{V}_r$ constitute orthonormal basis for $\mathcal{R}(A)$ and $\mathcal{R}(A^T)$, respectively. Consequently,

$$A_{r+1} = 0 \qquad (92)$$

and

$$A = \tilde{U}_r \tilde{D}_r \tilde{V}_r^T \qquad (93)$$

Starting from the idea of approximating the matrix A with matrix $\tilde{B}_k$, and the idea of representing all the images in a subspace of a smaller dimension, we propose a customized version of the algorithm (83)-(86), COD-A1 algorithm. The customization consists in a proper choice at each iteration for $\hat{u}_{i+1} \in \text{Range}\,(A_i)$ and $\hat{v}_{i+1} \in \text{Range}\left(A_i^T\right)$. We have tested other options for these initializations (each iteration), but the obtained results were not satisfactory.

Using this algorithm we try to approximate the matrix U (from the singular value decomposition of matrix $A = U\Sigma V^T$) with matrix $\tilde{U}_k$ from (83), which will serve as orthonormal basis for Range (A). Then we follow the classification process from the Eigenfaces algorithm: we compare the new expression of the picture we are looking for with all the columns of the new obtained matrix (dataset) and we seek the picture from the dataset that is closest (in a specific metric) to the image we are looking for.

Our proposed algorithm is the next one.

COD Algorithm - A1:

Step 1. Initialize $\hat{u}_1,\ \hat{u}_1 = \hat{u}_1 / \|\hat{u}_1\|$, $\hat{v}_1,\ \hat{v}_1 = \hat{v}_1 / \|\hat{v}_1\|$ *şi* $A_1 = A$.
Step 2. for $i = 1,\ 2,\ \ldots,\ k$
$\tilde{u}_i = A_i\hat{v}_i / \|A_i\hat{v}_i\|_2$
$\tilde{v}_i = A_i^T\hat{u}_i / \left\|A_i^T\hat{u}_i\right\|_2$
$\tilde{\sigma}_i = \left(\|A_i * \hat{v}_i\|_2 \left\|A_i^T\hat{u}_i\right\|_2\right) / \left(\hat{u}_i^T A_i \hat{v}_i\right)$
$A_{i+1} = A_i - \tilde{\sigma}_i\tilde{u}_i\tilde{v}_i^T$
reinitialize $\hat{u}_{i+1},\ \hat{u}_{i+1} = \hat{u}_{i+1} / \|\hat{u}_{i+1}\|$ *and* $\hat{v}_{i+1},\ \hat{v}_{i+1} = \hat{v}_{i+1} / \|\hat{v}_{i+1}\|$
end
Step 3. Let $U = [\tilde{u}_1\ \tilde{u}_2\ \ldots\ \tilde{u}_k]$ *and* $B = A_{k+1}$.
Step 4. Obtain $\Omega_i^T = [\omega_1^i,\ \omega_2^i, \ldots, \omega_k^i]$ *where* $col_i A = \sum_{j=1}^{k} \omega_j^i \tilde{u}_j$.
Step 5. Given a image Γ, *obtain* $\Gamma = \sum_{j=1}^{k} \omega_j \tilde{u}_j$.
Step 6. Represent Γ *as* $\Omega^T = [\omega_1,\ \omega_2, \ldots, \omega_k]$.
Step 7. Find $i_0 \in \{1, \ldots, k\}$ *satisfying* $\|\Omega - \Omega_{i_0}\| = \min_{1 \le i \le k} \|\Omega - \Omega_i\|$.

4.4 Singular value decomposition for tensors

In [21] L. Grecu (Liţă), E. Pelican, *A Low-Rank Tensor-Based Algorithm for Face Recognition* we propose another algorithm for face recognition, an algorithm based on tensors.

The motivation why tensors are used in face recognition is that often the data is stored as a tensor. Instead of storing all the data as a matrix, we store all photos belonging to the same person as a matrix, and all matrices coresponding to all persons form a tensor. This leads to a better ordering of the pictures from the dataset. In this ordering, the faces are classified into a

number of groups of different "expressions". This classification refers to photographic angles (e.g. left-portrait, right-portrait, front-portrait), illumination conditions (dark, lighted, etc.) or facial expressions (happy, sad, angry, etc.) and so on. For our dataset the "expressions" refers to lighting conditions.

In this section we present a generalization of the SVD theorem for matrices to SVD for tensors (see [8] and [13]).

Let $A \in I\!R^{l\times m\times n}$, $U \in I\!R^{l_0\times l}$ and $A \times_1 U$ a tensor of size $l_0 \times m \times n$, we have the following way of multiplication (1-mode tensor-matrix multiplication) $(A \times_1 U)(j, i_2, i_3) = \sum_{k=1}^{l} u_{j,k} a_{k,i_2,i_3}$. For $A \in I\!R^{l\times m\times n}$, $U \in I\!R^{m_0\times m}$ and $A \times_2 U$ a tensor of size $l \times m_0 \times n$ we have (2-mode tensor-matrix multiplication) $(A \times_2 U)(i_1, j, i_3) = \sum_{k=1}^{m} u_{j,k} a_{i_1,k,i_3}$, and for $A \in I\!R^{l\times m\times n}$, $U \in I\!R^{n_0\times n}$ and $A \times_3 U$ a tensor of size $l \times m \times n_0$ we have (3-mode tensor-matrix multiplication) $(A \times_3 U)(i_1, i_2, j) = \sum_{k=1}^{n} u_{j,k} a_{i_1,i_2,k}$. The i-mode and j-mode multiplication commute if $i \neq j,\ i,\ j \in \{1,\ 2,\ 3\} : (A \times_i U) \times_j V = (A \times_j V) \times_i U = A \times_i U \times_j V$. A tensor can be unfold into a matrix, $A_{(i)} = unfold_i(A)$, so:
$A_{(1)} = unfold_1(A) = (A(:,1,:)\ A(:,2,:)\ \ldots\ A(:,m,:))$,
$A_{(2)} = unfold_2(A) = \left(A(:,:,1)^T\ A(:,:,2)^T\ \ldots\ A(:,:,n)^T\right)$,
$A_{(3)} = unfold_3(A) = \left(A(1,:,:)^T\ A(2,:,:)^T\ \ldots\ A(l,:,:)^T\right)$.

In all unfoldings, row i of $A_{(j)}$ contains all the elements of A which have the j-th index equal to i. The inverse of the unfolding operation is folding $fold_i(unfold_i(A)) = A$. Using these unfoldings we obtain: $A \times_1 U = fold_1(U \cdot unfold_1(A))$, $A \times_2 U = fold_2(U \cdot unfold_2(A))$, and $A \times_3 U = fold_3(U \cdot unfold_3(A))$. We also introduce the scalar product $\langle A, B\rangle$ of two tensors $A,\ B \in I\!R^{l\times m\times n}$, as $\langle A, B\rangle = \sum_{i=1}^{l}\sum_{j=1}^{m}\sum_{k=1}^{n} A(i,j,k)\, B(i,j,k)$.

Theorem 8 *(HOSVD, see [8] and [13]) Tensor $A \in I\!R^{l\times m\times n}$ can be written as*

$$A = S \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)} \tag{94}$$

where $U^{(1)} \in I\!R^{l\times l}$, $U^{(2)} \in I\!R^{m\times m}$, $U^{(3)} \in I\!R^{n\times n}$ are orthogonal matrices. Matrices $U^{(i)}$ are obtained from $A_{(i)} = U^{(i)}\Sigma^{(i)}\left(V^{(i)}\right)^T$, $A_{(i)} = unfold_i(A)$, without forming the $V^{(i)}$ explicitly. $S = A \times_1 \left(U^{(1)}\right)^T \times_2 \left(U^{(2)}\right)^T \times_3 \left(U^{(3)}\right)^T$ is a tensor of the same size as A and satisfies: any two different slices fixed in the same mode are orthogonal (all-orthogonality) $\langle S(i,:,:), S(j,:,:)\rangle = 0,\ i \neq j$, $\langle S(:,i,:), S(:,j,:)\rangle = 0,\ i \neq j$, $\langle S(:,:,i), S(:,:,j)\rangle = 0,\ i \neq j$; and the norms of the slices along every mode are ordered, e.g., for the first mode we have $\|S(1,:,:)\| \geq \|S(2,:,:)\| \geq \ldots \geq 0$. The decomposition from (94) is not unique (see also [13] and [8])

Let $A \in I\!R^{n_i \times n_e \times n_p}$ be a tensor representing our training dataset and let a vector from $I\!R^{n_i}$ represent a picture from the testing set. Here n_i is the resolution of a picture ($n_i = M$ from previous sections), n_p is the number of persons from the dataset ($n_p = P$ from previous sections), and n_e is the number of expressions per person ($n_e \cdot n_p = N$ from previous sections). We want to find out if the algorithm identifies it correctly. In [13], for the algorithms from page 173-174 it is assumed that $n_i >> n_e n_p$. For a large dataset (e.g. ExtYaleB dataset), after reducing image size, we get $n_i << n_e n_p$. When performing the tests for ExtYaleB dataset with the algorithms from [13] we obtain smaller recognition rate than for PCA (Eigenfaces algorithm).

In order to address this issue we propose another algorithm for face recognition, when $n_i << n_e n_p$. For the problem under consideration, we use the following form of the HOSVD theorem $A = C \times_e G,\ C = S \times_i F \times_p H$, where $\times_i = \times_1,\ \times_e = \times_2,\ \times_p = \times_3$ şi $F = U^{(1)} \in I\!R^{n_i \times n_i},\ G = U^{(2)} \in I\!R^{n_e \times n_e},\ H = U^{(3)} \in I\!R^{n_p \times n_P}$ from HOSVD theorem.

For a given person p we have: $A(:,:,p) = C(:,:,p) \times_e G$. Tensors $A(:,:,p)$ and $C(:,:,p)$ are, in fact, matrices denoted by A_p and, respectively C_p. Hence $A_p(:,e)$ is the image of person p in expression e and the columns of matrix C_p are basis vectors for person p (let us denote this basis by person basis). It follows that $A_p = C_p G^T,\ p = 1, 2, \ldots, n_p$. Let $G^T = (g_1 \ldots g_{n_e})$, then $A_p(:,e) = C_p g_e$. Thus $g_e,\ e = 1, 2, \ldots, n_e$ are the coordinates of the image $A_p(:,e)$ of person p in expression e in the above mentioned basis.

Let $z \in I\!R^{n_i}$ be a picture from the testing set. We want to see if the picture is correctly identified. For this we have to solve $\min_{\alpha_p} \|C_p \alpha_p - z\|_2$. For each image z we have to solve n_p least square problems with $C_p \in I\!R^{n_i \times n_e}$. From $C = S \times_i F \times_p H$ we obtain that $C_p = F B_p$, where $B_p \in I\!R^{n_e n_p \times n_e},\ B_p = (S \times_p H)(:,:,p)$.

In order to reduce the computational effort we can truncate the tensors and matrices so that we obtain a truncated HOSVD decomposition for tensor A. Let k be the level of truncation and $F_k = F(:, 1:k)$ and we get $\hat{C} = (S \times_p H)(1:k,:,:) \times_i F_k$. Hence, we have to solve $\min_{\alpha_p} \left\| \hat{C}_p \alpha_p - z \right\|_2$. Our proposed algorithm is the next one, for$n_i << n_e n_p$.

Tensors Algorithm - A2:

Let z be the image we are looking for and choose k the level of truncation.
for $p = 1, 2, \ldots, n_p$
Let $\hat{C} = (S \times_p H)(1:k,:,:) \times_i F_k$
Solve $\min_{\alpha_p} \left\| \hat{C}_p \alpha_p - z \right\|_2$.
for $e = 1, 2, \ldots, n_e$
if $\|\alpha_p - g_e\|_2 < tol$, then is person p and STOP
end

end

Remark 7 *The level of truncation k can be chosen as in [42] or [10] or [14] and not empirically as in [13] (pages 116 and 173).*

5 Experiments

In this chapter we present experiments and results obtained with all algorithms figured in Chapters 1, 3 and 4. This chapter contains the following sections.

5.1 Experiments for the construction of the scanning matrix in Electromagnetic Geotomography

5.2 Experiments with SART algorithm

5.2.1 Paralelization of SART algorithm

5.3 Experiments with classes of algorithms for pattern recognition

5.3.1 Experiments with classes of algorithms for face recognition

5.3.2 Comparative study of algorithms NN, PCA and COD-A1 for pattern recognition

Bibliografie

[1] Andersen A. H., Kak A. C., *Simultaneous Algebraic Reconstruction Techniques (SART): a superior implementation of the ART algorithm*, Ultrasonic imaging, **6** (1984), 81–94.

[2] Björk A., *Numerical methods for least squares problems*, SIAM Philadelphia, 1996.

[3] Censor Y., Stavros A. Z., *Parallel optimization: theory, algorithms and applications*, Numer. Math. and Sci. Comp. Series, Oxford Univ. Press, New York, 1997.

[4] Chen J., Saad Y., *Lanczos Vectors versus Singular Vectors for Effective Dimension Reduction*, IEEE Transactions on Knowledge and Data Engineering, **21(8)** (2009), 1091-1103.

[5] Cimmino G., *Calcolo approssiomatto per le soluzioni dei sistemi di equazioni lineari*, Ric. Sci. progr. tecn. econom. naz. **1** (1938), 326-333.

[6] Dax A., *Orthogonalization Via Deflation: A Minimum Norm Approach for Low-Rank Approximations of a Matrix*, Journal of Computational and Applied Mathematics, **234(11)** (2010), 3091-3103.

[7] Dax A., *A minimum norm approach for low-rank approximations of a matrix*, SIAM Journal on Matrix Analysis and Applications, **30(1)** (2008), 236-260.

[8] De Lathauwer L., de Moor B., Vandewalle J., *A multilinear singular value decomposition*, SIAM Journal Matrix Anal. Appl, **21** (2000), 1252-1278.

[9] Delac K., Grgic M., Grgic S., *Independent Comparative Study of PCA, ICA, and LDA on the FERET Data Set*, International Journal of Imaging Systems and Technology, **15(5)** (2006), 252-260.

[10] Delac K., Grgic M., Liatsis P., *Appearance-based Statistical Methods for Face Recognition*, Proceedings of the 47th International Symposium ELMAR-2005 focused on Multimedia Systems and Applications, Zadar, Croatia, 2005, 151-158.

[11] Dines K. A., Lytle R. J., *Computerized geophysical tomography*, Proc. IEEE, **67** (1979), 1065-1073.

[12] Duda R.O., Hart P.E., Stork D.G., *Pattern Classification, second ed.*, Wiley-Interscience, New York, 2001.

[13] Elden L., *Matrix Methods in Data Mining and Pattern Recognition*, SIAM, Philadelphia, 2007.

[14] Glineur F., Lu L., Van Dooren P., Wang X., *Extended Lanczos Bidiagonalization for Dimension Reduction in Information Retrieval*, acceptată pentru publicare în Proceedings of the 8th International Conference on Natural Computation (ICNC'12), Maui, Hawaii, 2012.

[15] Golub G., Van Loan C., *Matrix Computation, 3rd Edition*, The John Hopkins University Press, Baltimore 1996.

[16] Grecu L., Popa C., *Constrained SART algorithm for invers problems in image reconstruction*, Inverse Problems and Imaging, 1(7) (2013), 199-216.

[17] Grecu L., Nicola A., *Some results on simultaneous algebraic techniques in image reconstruction from projections*, ROMAI Journal, 2 (2009), 79-96, http://rj.romai.ro/

[18] Grecu (Liţă) L., *On the construction of the scanning matrix in Electromagnetic Geotomography*, Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara, Seria Matematică - Fizică, 1 (2012), 42-56.

[19] Grecu (Liţă) L. , Pelican E., *Systematic and comparative experiments with some algorithms for pattern recognition*, acceptată pentru publicare în volumul conferinţei Nineth Workshop on Mathematical Modelling of Environmental and Life Sciences Problems, November 1-4, 2012, Constanţa, Romania.

[20] Grecu (Liţă) L. , Pelican E., *Customized Orthogonalization via Deflation Algorithm with Applications in Face Recognition*, trimisă spre publicare la Carpathian Journal of Mathematics, 2013.

[21] Grecu (Liţă) L. , Pelican E., *A Low-Rank Tensor-Based Algorithm for Face Recognition*, trimisă spre publicare la Pattern Recognition Letters, 2013.

[22] Guo G.-D., Zhang H.-J., Li S.Z., *Pairwise Face Recognition*, Proc. of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, 2 (2001), Vancouver, Canada, 282-287.

[23] Guo G.-D., Zhang H.-J., *Boosting for Fast Face Recognition*, Second International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems, RATFG-RTS'01, Vancouver, Canada, 96-100.

[24] Herman G. T., *Image reconstruction from projections. The fundamentals of computerized tomography*, Academic Press, New York, 1980.

[25] Horn R. A., Johnson C. R., *Matrix analysis*, Cambridge University Press, New York, 1990.

[26] Hotelling H., *Analysis of a complex of statistical variables into principal components*, J. Educ. Psychol., **24** (1933), 417-441, 498-520.

[27] Jiang M., Wang G., *Convergence of the simultaneous algebraic reconstruction technique (SART)*, in Proc. 35th Asilomar Conf. Signals, Systems, and Computers, Pacific Grove, CA, 2001, 360-364.

[28] Jiang M., Wang G., *Convergence studies on iterative algorithms for image reconstruction*, IEEE Trans. on Medical Imaging, **22** (2003), 569-579.

[29] Jolliffe I.T. , *Principal Component Analysis*, Springer-Verlag, Second Edition, New York, 2002.

[30] Kaczmarz S., *Angenaherte Auflosung von Systemen linearer Gleichungen*, Bull. Acad. Polonaise Sci. et Lettres **A** (1937), 355-357.

[31] Kirby M., Sirovich L., *Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **12(1)** (1990), 103-108.

[32] Koltracht I., Lancaster P., Smith D., *The structure of some matrices arising in tomography*, Linear algebra and its applications, **130** (1990), 193-218.

[33] Koltracht I., Lancaster P., *Constraining Strategies for Linear Iterative Processes*, IMA Journal of Numerical Analysis, **10** (1990), 555-567.

[34] Martinez A.M., Kak A.C., *PCA versus LDA*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **23(2)** (2001), 228-233.

[35] Meyer C. D., *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, 2000.

[36] Murty M. Narasimha, Devi V. Susheela, *Pattern Recognition. An Algorithmic Approach*, Springer, First Edition, London, 2011.

[37] Natterer F., *The Mathematics of Computerized Tomography*, John Wiley and Sons, New York, 1986.

[38] Nicola A., Petra S., Popa C. and Schnörr C., *On a general extending and constraining procedure for iterative methods*, Intern. Journal of Computer Mathematics, **89(2)** (2012), 231-253.

[39] Pan X., Sidky E. Y., Vannier M., *Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction?*, Inverse Problems, **25(12)** (2009).

[40] Pearson K., *On lines and planes of closest fit to systems of points in space*, Phil. Mag., **2(6)** (1901), 559-572.

[41] Pelican E., Grecu L., *Comparison Between Some Matrix Methods with Applications in Pattern Recognition*, prezentată la Applied Linear Algebra Conference, May 24-28, 2010, Novi Sad.

[42] Pelican E., Grecu L., *Low-Rank Matrix Methods in Pattern Recognition*, prezentată la Balkan Conference on Operational Research (BALCOR), 2009, publicată electronic cu ISBN: 973-86979-9-9.

[43] Pelican E., Grecu L., *Solving the Pattern Recognition Problem with some Low-Rank Approximation Based Algorithms*, prezentată la XIème Colloque Franco-Roumain de Mathématiques Appliquées, 24-30 August, 2012, Bucureşti.

[44] Pentland A., Starner T., Etcoff N., Masoiu A., Oliyide O., and Turk M., *Experiments with Eigenfaces*, prezentată la Looking at People Workshop, IJCAI '93, Franţa, 1993.

[45] Popa C., *Projection algorithms, classical results and developments. Applications to image reconstructions*, Lambert Academic Publishing - AV Akademikerverlag GmbH & Co. KG, Saarbrucken, Germany, 2012.

[46] Popa C., *Constrained Kaczmarz extended algorithm for image reconstruction*, Linear Algebra and its Applications, **429** (2008), 2247-2267.

[47] Popa C., *Extensions of block-projections methods with relaxation parameters to inconsistent and rank-defficient least-squares problems*; B I T, **38(1)** (1998), 151-176.

[48] Popa C., *A hybrid KaczmarzConjugate Gradient algorithm for image reconstruction*, Mathematics and Computers in Simulation, **80** (2010), 2272-2285.

[49] Sirovich L., Kirby M., *Low-dimensional Procedure for the Characterization of Human Faces*, Journal of the Optical Society of America A - Optics, Image Science and Vision, **4(3)** (1987), 519-524.

[50] Tanabe K., *Projection Method for Solving a Singular System of Linear Equations and its Applications*, Numer. Math., **17** (1971), 203-214.

[51] Turk M., Pentland A., *Eigenfaces for Recognition*, Journal of Cognitive Neuroscience, **1** (1991), 71-86.

[52] Turk M., Pentland A., *Face Recognition using Eigenfaces* , Computer Vision and Pattern Recognition Proceedings CVPR '91, 1991, 586-591.

[53] http://www.math.ucf.edu/˜xli/SVD_PCA.pdf