Embedding of the Extended Euclidean Distance into Pattern Recognition with Higher-Order Singular Value Decomposition of Prototype Tensors

Bogusław Cyganek

AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland
cyganek@agh.edu.pl

Abstract. The paper presents architecture and properties of the ensemble of the classifiers operating in the tensor orthogonal spaces obtained with the Higher-Order Singular Value Decomposition of prototype tensors. In this paper two modifications to this architecture are proposed. The first one consists in embedding of the Extended Euclidean Distance metric which accounts for the spatial relationship of pixels in the input images and allows robustness to small geometrical perturbations of the patterns. The second improvement consists in application of the weighted majority voting for combination of the responses of the classifiers in the ensemble. The experimental results show that the proposed improvements increase overall accuracy of the ensemble.

Keywords: Pattern classification, ensemble of classifiers, Euclidean Distance, IMED, HOSVD

1 Introduction

This paper is an extension of our previous work on development of the image classification with the ensemble of tensor based classifiers [4]. The method showed to be very robust in terms of accuracy and execution time, since many existing methods do not account for the multi-dimensionality of the classified data [19][21][22].

Processing and classification of the multi-factor dependent data can be addressed with help of methods operating with tensors and their decompositions. One of the pioneered methods from this group is the face recognition system, coined tensor-faces, proposed by Vasilescu and Terzopoulos [22]. In their approach tensors are proposed to cope with multiple factors of face patterns, such as different poses, views, illuminations, etc. Another tensor based method for handwritten digits recognition was proposed by Savas *et al.* [17][15]. Their method assumes tensor decomposition which allows representation of a tensor as a product of its core tensor and a set of unitary mode matrices. This decomposition is called Higher-Order Singular Value Decomposition (HOSVD) [1][14][11]. A similar approach was undertaken by Cyganek in the system for road signs recognition [3]. In this case, the input pattern tensor is built from artificially generated deformed versions of the prototype road sign exemplars. All aforementioned systems, which are based on HOSVD, show very high

accuracy and high speed of response. However, computation of the HOSVD from large size tensors is computationally demanding since the algorithm requires successive computation of the SVD decompositions of matrices obtained from tensor flattening in different modes [13]. In practice, these matrices can be very large since they correspond to the products of all dimensions of the input tensor. In many applications this can be very problematic. To overcome this problem an ensemble with smaller size pattern tensor was proposed by Cyganek [4]. In the proposed methods tensors are of much smaller size than in a case of a single classifier due to the bagging process. However, despite the computational advantages, the proposed ensemble based method shows better accuracy when compared to a single classifier.

In this paper two modifications to the previously presented method are proposed. The first one is embedding of the Extended Euclidean Distance metric, recently introduced by Wang *et al.* [23]. This allows robustness to small geometrical perturbations of the input patterns since the new metric accounts for the spatial relationship of pixels in the input images. The second improvement consists in application of the weighted majority voting for combination of the responses of the classifiers in the ensemble. The experimental results show that in many cases the proposed improvements allow an increase of the overall accuracy of classification.

The rest of the paper is organized as follows. In section 2 properties of the Euclidean Image Distance are presented. In Section 3 the architecture of the proposed ensemble of the HOSVD multi-classifiers is discussed. Pattern recognition by the ensemble of the tensor classifiers is discussed in section 4. Experimental results are presented in section 5. The paper ends with conclusions in section 6.

2 Embedding Euclidean Image Distance

Images are 2D structures in which a scalar, vector (color) or multi-dimensional (MRI) value of a pixel is as important as its position within image coordinate space. However, the second aspect is not easy to be accounted for due to geometrical transformation of images of observed objects. On the other hand, image recognition heavily relies on comparison of images for which the Euclidean metric is the most frequently used one, mostly due to its popularity and simplicity in computations. However, Wang *et al.* proposed a better metric than Euclidean which takes into account also spatial relationship among pixels [23]. The proposed metric, called IMage Euclidean Distance (IMED), shows many useful properties, among which the most important is its insensitivity to small geometrical deformations of compared images.

More specifically, instead of the Euclidean metric between the two images $\mathbf{X}$ and $\mathbf{Y}$ of dimensions $M \times N$ each, given as follows

$$D_E\left(\mathbf{X},\mathbf{Y}\right) = \sum_{k=1}^{MN}\left(x^k - y^k\right)^2 = \left(\mathbf{x}-\mathbf{y}\right)^T\left(\mathbf{x}-\mathbf{y}\right), \qquad (1)$$

Wang *et al.* propose to use the following extended version

$$D_G\left(\mathbf{X},\mathbf{Y}\right)=\sum_{k,l=1}^{MN} g_{kl}\left(x^k-y^k\right)\left(x^l-y^l\right)=\left(\mathbf{x}-\mathbf{y}\right)^T\mathbf{G}\left(\mathbf{x}-\mathbf{y}\right), \tag{2}$$

where **x** and **y** are column vectors formed by the column- or row-wise vectorization of the images **X** and **Y**, respectively, and g_{kl} are elements of the symmetric nonnegative matrix **G** of dimensions $MN\times MN$, which defines the metric properties of the image space.

Thanks to the above formulation, information on spatial position of pixels can be embedded into the distance measure, through the coefficients g_{kl}. In other words, the closer the pixels are, the higher value of g_{kl} should be, reaching its maximum for $k=l$. The distance between pixel positions (not values) is defined on an integer image lattice simply as a function of the 'pure' Euclidean distance between the points, as follows

$$g_{kl}=f\left(\left|\mathbf{P}_k-\mathbf{P}_l\right|\right)=\frac{1}{2\pi\sigma^2}e^{-\frac{\left|\mathbf{P}_k-\mathbf{P}_l\right|^2}{2\sigma^2}}, \tag{3}$$

where $\mathbf{P}_i=\left[p_i^1,p_i^2\right]^T$ denotes position of the i-th pixel in the image, while σ is a width parameter, usually set to 1 [23]. Finally, incorporating (**3**) into (**2**) the IMED distance among image **X** and **Y** is obtained, as follows

$$D_{IMED}\left(\mathbf{X},\mathbf{Y}\right)=\frac{1}{2\pi\sigma^2}\sum_{k,l=1}^{MN} e^{-\frac{\left(p_k^1-p_l^1\right)^2+\left(p_k^2-p_l^2\right)^2}{2\sigma^2}}\left(x^k-y^k\right)\left(x^l-y^l\right). \tag{4}$$

The D_{IMED} image metric given in (**4**) can be used for a direct comparison of images, such as in the case of the k-nearest neighbor method, etc. It can be also incorporated into other classification algorithms, such as the discussed HOSVD. This can be achieved substituting D_{IMED} into all places in which the D_E was used.

However, for large databases of images direct computation of (**4**) can be expensive. An algorithm to overcome this problem was proposed by Sun *et al.* after observing that computation of D_{IMED} can be equivalently stated as a transform domain smoothing [18]. They developed the Convolution Standardized Transform (CST) which approximates well the D_{IMED} . For this purpose the following separable filter was used

$$\mathbf{H}=\mathbf{h}\otimes\mathbf{h}^T=\mathbf{h}\mathbf{h}^T, \tag{5}$$

where $\otimes$ denotes the Kronecker product of two vectors **h** and $\mathbf{h}^T$ with the following components

$$\mathbf{h}=\begin{bmatrix}0.0053 & 0.2171 & 0.5519 & 0.2171 & 0.0053\end{bmatrix}^T. \tag{6}$$

The filter **h** given by (**6**) was also used in our computations since it offers much faster computations than direct application of (**4**).

Fig. 1 shows examples of application of the Standardizing Transformation for selected pictograms of the road signs in implementation with the filter **h** in (**6**). It is visible that ST operates as a low-pass filter (lower row). This way transformed patterns are fed to the classifier system.

Fig. 1. Visualization of Standardizing Transform applied to the road sign pictograms. Original pictograms (upper row). After transformation (lower row).

Finally, it should be noticed that the IMED transformation should not be confused with the Mahalanobis distance or the whitening transformation [6][5]. Specifically, in equation (**2**) we do not assume computation of any data distribution nor probabilistic spaces. In other words, the main difference lies in definition of the matrix **G** in (**2**) which elements, given by (**3**), convey information on mutual positions of the points. In contrast, for the Mahalanobis distance **G** would be an inverse of the covariance matrix which elements are computed directly from the values of **x** and **y** disregarding their placement in the images.

3 Architecture of the Ensemble of HOSVD Multi-Classifiers

Multidimensional data are handled efficiently with help of the tensor based methods since each degree of freedom can be represented with a separate index of a tensor [1][2]. Following this idea, multidimensional training patterns can be efficiently represented by a prototype tensor [3]. For the purpose of pattern recognition the prototype patterns tensor can be further decomposed into the orthogonal components which span a prototype tensor space. For the decomposition the Higher-Order Singular Value Decomposition can be used [1][13][11]. This way obtained orthogonal bases are then used for pattern recognition in a similar way to the standard PCA based classifiers [5][20][21].

The HOSVD method allows any P-dimensional tensor $\mathcal{T} \in \Re^{N_1 \times N_2 \times \ldots N_m \times \ldots N_n \times \ldots N_P}$ to be equivalently represented in the following form [13][14]

$$\mathcal{T} = \mathcal{Z} \times_1 \mathbf{S}_1 \times_2 \mathbf{S}_2 \ldots \times_P \mathbf{S}_P , \tag{7}$$

where $\mathbf{S}_k$ are $N_k \times N_k$ unitary mode matrices, $\mathcal{Z} \in \Re^{N_1 \times N_2 \times \ldots N_m \times \ldots N_n \times \ldots N_P}$ is a core tensor. $\mathcal{Z}$ fulfills the following properties [13][14]:

1. (Orthogonality) Two subtensors $\mathcal{Z}_{n_k=a}$ and $\mathcal{Z}_{n_k=b}$ for all possible values of k for which $a \neq b$ it holds that

$$\mathcal{Z}_{n_k=a} \cdot \mathcal{Z}_{n_k=b} = 0 . \tag{8}$$

2. (Energy) All subtensors of $\mathcal{Z}$ for all k can be ordered according to their Frobenius norms, as follows

$$\left\| \mathcal{Z}_{n_k=1} \right\| \geq \left\| \mathcal{Z}_{n_k=2} \right\| \geq \ldots \geq \left\| \mathcal{Z}_{n_k=N_P} \right\| \geq 0 , \tag{9}$$

The a-mode singular value of $\mathcal{T}$ is defined as follows

$$\left\| \mathcal{Z}_{n_k=a} \right\| = \sigma_a^k . \tag{10}$$

An algorithm for computation of the HOSVD is based on successive computations of the SVD decomposition of the matrices composed of the flattened version of the tensor $\mathcal{T}$. The algorithm requires a number of the SVD computations which is equal to the valence of that tensor. The detailed algorithm can be referred to in the literature [1][13][14].

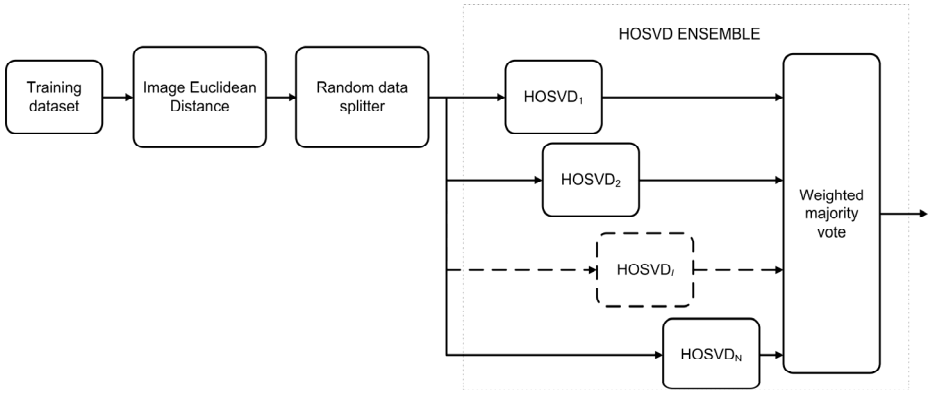

Fig. 2. Architecture of the ensemble with the HOSVD classifiers. Data preprocessed with the Image Euclidean Transformation. Bagging method used for training. Outputs are combined with the weighted majority voting.

Fig. 2 shows architecture of the proposed ensemble of the HOSVD classifiers. All training and testing data are preprocessed with the Image Euclidean Transformation

described in section (2). Then, each HOSVD is trained with only a partition of the training dataset obtained in the bagging process.

In the next step, accuracies of each of the classifiers in the ensemble are assessed using the *whole* training dataset. These are then used to compute the weights of the classifiers in the ensemble. In the run-time their outputs are combined with the weighted majority voting scheme, as will be described in the next section.

4 Pattern Recognition by the Ensemble of Tensor Classifiers

It can be observed that thanks to the commutative properties of the *k*-mode tensor multiplication [17], the following sum can be constructed for each mode matrix $\mathbf{S}_i$ in (7)

$$\mathcal{T} = \sum_{h=1}^{N_P} \mathcal{T}_h \times_P \mathbf{s}_P^h \,. \tag{11}$$

In the above the tensors

$$\mathcal{T}_h = \mathcal{Z} \times_1 \mathbf{S}_1 \times_2 \mathbf{S}_2 \ldots \times_{P-1} \mathbf{S}_{P-1} \tag{12}$$

form the basis tensors, whereas $\mathbf{s}^h{}_P$ denote columns of the unitary matrix $\mathbf{S}_P$. Because each $\mathcal{T}_h$ is of dimension P-1 then $\times_P$ in (11) is an outer product, i.e. a product of two tensors of dimensions P-1 and 1. Moreover, due to the orthogonality properties (**8**) of the core tensor $\mathcal{Z}$ in (12), $\mathcal{T}_h$ are also orthogonal. Hence, they can constitute a basis which spans a subspace. This property is used to construct a HOSVD based classifier.

In the tensor space spanned by $\mathcal{T}_h$, pattern recognition can be stated as a measuring a distance of a given test pattern $\mathbf{P}_x$ to its projections into each of the spaces spanned by the set of the bases $\mathcal{T}_h$ in (12). This can be written as the following minimization problem [17]

$$\min_{i, c_h^i} \underbrace{\left\| \mathbf{P}_x - \sum_{h=1}^{H} c_h^i \mathcal{T}_h^i \right\|^2}_{Q_i} , \tag{13}$$

where the scalars $c^i{}_h$ denote unknown coordinates of $\mathbf{P}_x$ in the space spanned by $\mathcal{T}_h^i$, $H \leq N_P$ denotes a number of chosen dominating components.

To solve (13) the squared norm Q of (13) is created for a chosen index i. Assuming further that $\mathcal{T}_h^i$ and $\mathbf{P}_x$ are normalized the following is obtained (the *hat* mark indicates tensor normalization)

$$\rho_i = 1 - \sum_{h=1}^{H} \left\langle \hat{\mathcal{T}}_h^i, \hat{\mathbf{P}}_x \right\rangle^2 . \tag{14}$$

Thus, to minimize (13) the following value needs to be maximized

$$\hat{\rho}_i = \sum_{h=1}^{H} \left\langle \hat{\boldsymbol{\mathcal{T}}}_h^i, \hat{\mathbf{P}}_x \right\rangle^2 , \tag{15}$$

Thanks to the above, the HOSVD classifier returns a class i for which its ρ_i from (15) is the largest.

Table 1. Structure of a matrix of partial accuracies for each classifier and each training prototype pattern

Digit	0	1	2	3	4	5	6	7	8	9
$HOSVD_0$	p_{00}	p_{00}	p_{00}	…						
$HOSVD_1$	p_{10}	p_{11}	p_{12}	…						
$HOSVD_2$	p_{20}	p_{21}	p_{22}	…						
…	…	…	…	…						

In this work also different fusion methods were tested. Especially, the majority voting scheme was substituted for the weighted majority vote [10][16]. As alluded to previously, we proposed to use bagging to train the HOSVD classifiers from the ensemble which allows efficient memory usage. However, the partitions used for bagging contain less exemplars than all available for each prototype pattern. Therefore we further propose to use the whole training dataset to test each classifier trained with only fraction of that dataset for recognition of each pattern. This way we can assign some weight accuracies p_{kl} for each classifier k and for each trained class l. These are defined as follows

$$p_{kl} = \frac{N_{TP}^l}{N_{TP}^l + N_{FP}^l} , \tag{16}$$

where N_{TP}^l denotes a number of true positive responses and N_{FP}^l false positives, respectively. Table 1 visualizes this process for ten training patterns, such as digits.

Further, it is assumed that the classifiers are independent and each is endowed with its individual accuracies p_{kl}. If their outputs are combined with the weighted majority voting scheme, then accuracy of the ensemble is maximized by assigning the following weights [12][9]

$$b_{kl} = \log \frac{p_{kl}}{1 - p_{kl}} , \tag{17}$$

where p_{kl} are given by (**16**). On the other hand, for a test pattern $\mathbf{P}_x$ each of the HOSVD classifiers in the ensemble responds with its class and assigned vote strength, as follows

$$d_{kl} = \begin{cases} \hat{\rho}_{kl}, & if\ \ HOSVD_k\ labels\ class\ l \\ 0, & otherwise \end{cases} , \tag{18}$$

where $\hat{\rho}_{kl}$ denotes a maximal value of $\hat{\rho}_i$ in (**15**) and for the l-th classifier in ensemble and for pattern class $k=i$. Finally, the following discriminating function is computed

$$g_k\left(\hat{\mathbf{P}}_x\right) = \log\left(P_k\right) + \sum_{l=1}^{L} d_{kl} b_{kl}\,, \tag{19}$$

where P_k denotes the prior probability for the k-th class. However, the latter is usually unknown, so in the rest of experiments the first term in (**19**) was set to 0.

5 Experimental Results

The presented method was implemented in C++ using the HIL library [2]. Experiments were run on the computer with 8 GB RAM and Pentium® Quad Core Q 820 (clock 1.73 GHz).

For the experiments the USPS dataset was used [8][24]. The same set was also used by Savas *et al.* [17], as well as in the paper [4]. This dataset contains selected and preprocessed scans of the handwritten digits from envelopes of the U.S. Postal Service. Fig. 3 depicts some digits from the training and from the testing sets, respectively. The dataset is relatively difficult for machine classification since the reported human error is 2.5%. Therefore it has been used for comparison of different classifiers [15][17]. Originally the test and train patterns from the ZIP database come as the 16×16 gray level images.

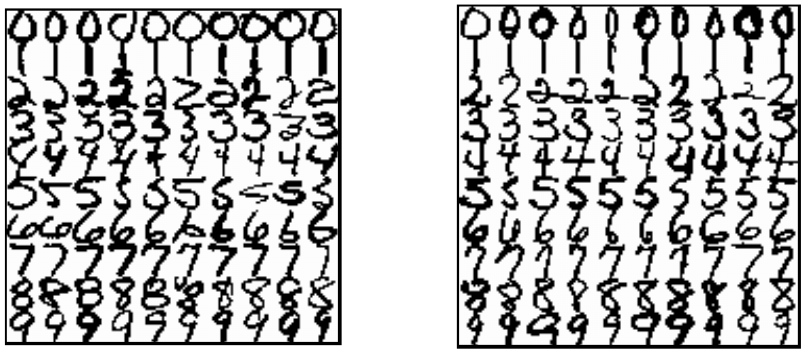

Fig. 3. Two data sets from the ZIP database. Training set (a), and testing set (b)

The database is divided into the training and testing partitions, counting 7291 and 2007 exemplars, respectively.

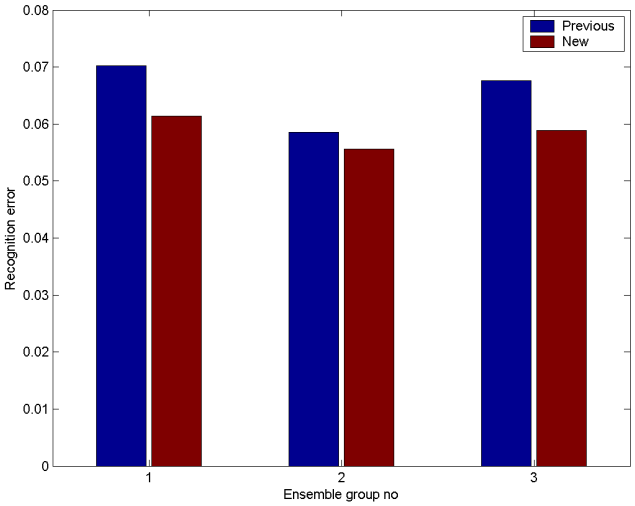

Fig. 4. Comparison of the accuracies of the three different ensemble settings without and with modifications proposed in this paper (the lower, the better). Blue bars for reference settings from [4]. Red bars relate to the method proposed in this paper, i.e. with the IMED and weighted majority voting.

Each experimental setup was run number of times and an average answer is reported. In all cases the Gaussian noise was added to the input image at level of 10%, in accordance with the procedure described in [2]. An analysis of different combinations of data partitions, number of classifiers in ensemble, number of dominating components, as well as input image size is discussed in our previous work [4]. In this paper we choose the best settings described in experimental results of the mentioned paper [4] and tested influence of the new IMED based preprocessing method, as well as new output fusion method. Results for three different settings are shown in the bar graph in Fig. 4.

For the experimental setups in this work were chosen three best setups from our previous work [4]. These are summarized in Table 2.

Table 2. Three best experimental setups of the ensemble from [4]

No. \ Param.	Number of experts	Data in samples	Important components	Image resolution	Noise [%]
1	11	64	16	16x16	10
2	15	192	16	32x32	10
3	33	64	16	16x16	10

The first experiments were run with configurations of the ensembles from Table 2. Then new propositions of IMED and weighted majority voting were introduced and run again. Each experiment setup was run 25 times and average parameters are reported. In all cases the proposed modifications allowed better results of 0.2-1% with negligible time penalty due to separability of the IMED filter.

6 Conclusions

In this paper an extended version of our previous work on image classification with the ensemble of tensor based classifiers is presented [4]. In this method, thanks to the construction of the ensemble of cooperating classifiers, tensors are of much smaller size than in a case of a single classifier. Each classifier in this ensemble is trained with data partition obtained from bagging. Such approach allows computations with much smaller memory requirements. However, the method shows also better accuracy when compared to a single classifier. In this paper two modifications to this formulation were discussed. The first is to apply input pattern preprocessing with embedding of the Extended Euclidean Distance metric. This allows robustness to small geometrical perturbations thanks to the metric which accounts for the spatial relationship of pixels in the input images. The second improvement consists in application of the weighted majority voting for combination of the responses of the classifiers in the ensemble. The experimental results show that in many cases the proposed improvements allow an increase of the overall accuracy of classification in order of 0.2-1%. The method is highly universal and can be used with other types of patterns.

Acknowledgement

The work was supported in the years 2011-2012 from the funds of the Polish National Science Centre NCN, contract no. DEC-2011/01/B/ST6/01994.

References

1. Cichocki, A., Zdunek, R., Amari, S.: Nonnegative Matrix and Tensor Factorization. IEEE Signal Processing Magazine, Vol. 25, No. 1, 142-145 (2008)
2. Cyganek, B., Siebert, J.P.: An Introduction to 3D Computer Vision Techniques and Algorithms, Wiley (2009)
3. Cyganek, B.: An Analysis of the Road Signs Classification Based on the Higher-Order Singular Value Decomposition of the Deformable Pattern Tensors, Advanced Concepts for Intelligent Vision Systems Acivs 2010, LNCS 6475, Springer, 191–202, (2010)
4. Cyganek, B.: Ensemble of Tensor Classifiers Based on the Higher-Order Singular Value Decomposition. HAIS 2012, Springer, Part II, LNCS 7209, 578–589 (2012)
5. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley (2001)
6. Fukunaga, K.: Introduction to Statistical Pattern Recognition. 2nd Edition. Academic Press (1990)

7. Grandvalet, Y.: Bagging equalizes influence. Machine Learning, Vol. 55, 251-270 (2004)
8. Hull, J.: A database for handwritten text recognition research. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 5, 550–554 (1994)
9. Jackowski, K., Woźniak, M.: Algorithm of designing compound recognition system on the basis of combining classifiers with simultaneous splitting feature space into competence areas. Pattern Analysis and Applications 12, 415-425 (2009)
10. Kittler, J., Hatef, M., Duing, R.P.W., Matas, J.: On Combining Classifiers. IEEE PAMI, Vol. 20, No. 3, 226-239 (1998)
11. Kolda, T.G., Bader, B.W.: Tensor Decompositions and Applications. SIAM Review, 455-500 (2008)
12. Kuncheva, L.I.: Combining Pattern Classifiers. Methods and Algorithms. Wiley Interscience (2005)
13. Lathauwer, de L.: Signal Processing Based on Multilinear Algebra. PhD dissertation, Katholieke Universiteit Leuven (1997)
14. Lathauwer, de L., Moor de, B., Vandewalle, J.: A Multilinear Singular Value Decomposition. SIAM Journal of Matrix Analysis and Applications, Vol. 21, No. 4, 1253-1278 (2000)
15. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-Based Learning Applied to Document Recognition. Proc. IEEE on Speech & Image Processing. Vol. 86, No. 11, 2278-2324 (1998)
16. Polikar, R.: Ensemble Based Systems in Decision Making. IEEE Circuits and Systems Magazine. pp. 21-45 (2006)
17. Savas, B., Eldén, L.: Handwritten digit classification using higher order singular value decomposition. Pattern Recognition, Vol. 40, 993-1003 (2007)
18. Sun, B., Feng, J.: A Fast Algorithm for Image Euclidean Distance. Chinese Conference on Pattern Recognition CCPR '08, 1-5 (2008)
19. Szeliski, R.: Computer Vision. Algorithms and Applications. Springer (2011)
20. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. 4th ed., Academic Press (2009)
21. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience 3 (1), 71–86 (1991)
22. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear analysis of image ensembles: TensorFaces. European Conference on Computer Vision, Denmark, LNCS 2350, Springer, 447-460 (2002)
23. Wang, L., Zhang, Y., Feng, J.: On the Euclidean Distances of Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, 1334-1339 (2005)
24. *www-stat.stanford.edu/~tibs/ElemStatLearn/*