Utility Functions as Aggregation Functions in Face Recognition

Michał Dolecki, Paweł Karczmarek[1], Adam Kiersztyn
Institute of Mathematics and Computer Science
The John Paul II Catholic University of Lublin
ul. Konstantynów 1H, 20-708, Lublin, Poland
{michal.dolecki, pawelk, adam.kiersztyn}@kul.pl

Witold Pedrycz
Department of Electrical & Computer Engineering
University of Alberta
Edmonton T6R 2V4 AB, Canada
Department of Electrical and Computer Engineering
King Abdulaziz University
Jeddah, 21589, Saudi Arabia
Systems Research Institute
Polish Academy of Sciences
Warsaw, Poland
wpedrycz@ualberta.ca

***Abstract*—Face recognition by computers in recent years has been a topic of intensive studies. In this problem, we witness several challenges: one has to cope with large data sets, solve problems of data extraction, and deal with poor quality of images caused by e.g., poor lighting of the subject. There have been a lot of algorithms and classifiers developed, which are aimed at recognizing faces of individuals. In this paper, we present a novel classification method, which involves a collection of classifiers with a certain utility function regarded as an aggregation operator. The nearest neighbor method with various similarity measures is used as a generic classifier for selected face areas. The main task is to assign photos of a person to one of the classes of image present in the available database. This problem is similar to the decision-making process with some evident analogies. If in face recognition, a single classifier is being used, the problem becomes similar to the one of decision-making with a single criterion. When having several classifiers, the problem resembles a problem of a multi-criteria decision making. The second scenario requires an aggregation of the results produced by different classifiers. The paper presents the use of the utility function which is well-known in the decision-making theory as an aggregation operator applied to the results of various classifiers. The study is focused on the two-factor utility function and its variants.**

Keywords—face recognition; aggregation functions; utility functions

I. Introduction

Computer vision has been an important research field in information technology in recent years [1, 2]. There have been developed numerous methods and algorithms to carry out this task, particularly in face recognition. It is important to note that the model of face recognition exhibits some resemblance or coincides with the models available in the area of decision support systems. Face recognition system receives an image of human's face and has to decide to which of the photos (individuals) coming from the existing database this image is the most similar to. This situation is similar to the typical decision making and forms a starting point to cast the face recognition problem in the setting of the decision making theory.

Let us recall that decision problem concerns a situation where some agent (decision maker) is exposed to several alternatives (say, actions) and he/she has to select only one of them. This situation is simple when we can assign to possible decisions some numbers representing the decision maker's profit. In this scenario, each option is evaluated according to a single criterion. Now, it is sufficient to evaluate each option and choose the one that maximizes the utility function. More interesting problem arises in a situation in which there are several criteria being used to evaluate possible decisions (outcomes). Any existing criterion produces an assessment for each alternative. Therefore, having n criteria, this yields an n-element vector of ratings obtained for each of the m alternatives. Finally, one has to introduce the ordering relation in this set, so that it is possible to come up with a final decision. Systems operating according to this specific one or other similar schemes are commonly considered and here decision support systems dedicated to business negotiation or production could be mentioned, as some representative examples. The decision-maker (viz. human, algorithm, or computer program) evaluates each of the possible decisions using his/her set of criteria. In the case where each option is analyzed using the m-element set of criteria it is obtained a vector with m numbers being the rankings of that option. The evaluation of a single value for each assessments vector can be performed using aggregation functions.

This study identifies an interesting link between the face recognition algorithms and decision support methods. The aim of this study is to propose new aggregation functions based on utility function and examine their performance in face recognition problems. This research has been inspired by similarities identified between face recognition and decision making problem. We analyze different variants of the commonly used utility functions and view them here as an

[1] Corresponding author
The authors are supported by National Science Centre, Poland (grant no. 2014/13/D/ST6/03244). Support from the Canada Research Chair (CRC) program and Natural Sciences and Engineering Research Council is gratefully acknowledged (W. Pedrycz).

aggregation function of the results produced by specific classifiers. There are used two factorial utility functions, which are based on the well-known Cobb-Douglas function [3]. In addition, there is analyzed it's modification partly described in [4-6]. A novel idea presented in this work concerns a reduction of typical computer vision and classification problem to the problem of multi-criteria decision-making theory with an application of new aggregation function built on the basis of two factor utility function. The outcomes produced of such methods are tested on two publicly available databases containing facial images, namely AT&T [7] (formerly known as ORL) and FERET [8].

The paper is organized as follows. Section II offers a brief introduction to face recognition and aggregation functions. In Section III, we focus on the details of experiments and discuss the obtained results. In Section IV conclusions and perspectives of future work are covered.

II. Face Recognition and Aggregation Functions

In face recognition methods, we can distinguish two main groups, namely holistic and feature-based matching [9]. Algorithms in the first group use the data concerning the whole face. In this group, one can refer to Eigenfaces (PCA) [10], Fisherfaces (LDA) [11], Support Vector Machines (SVM) [12], deep learning [13], etc. Principal Component Analysis (PCA) is one of the statistical analysis methods falling under this category. Given N data set of with dimensionality M, each of them can be interpreted as a collection of N points of M-dimensional space. The aim of the PCA is to transform the coordinate system of this space to maximize the variance of first coordinates, then the variance of the second coordinates, and so on. The transformed coordinates are called the principal components. The new space contains points in which most of the variability is explained by a few factors only. PCA is often used to reduce the data dimensionality. Linear Discriminant Analysis (LDA) is used in machine learning to form the linear combination of features that lead to the best discrimination between two or more classes of objects. The resulting combinations are used as a linear classifier or to carry out further dimensionality reduction. These algorithms are commonly used in face recognition. A digital face image consisting of a very large number of pixels is reduced to a smaller set of linear combinations, which can then be used for classification purposes. The linear combinations of features of the facial image obtained using the LDA are referred to as Fisherfaces while the PCA reduction results in a collection of eigenfaces.

The second group of face recognition algorithms includes methods using specific facial features like eyes, eyebrows, nose, or mouth. This group contains, among others, Elastic Bunch Graph Matching (EBGM) [14], geometrical methods [15], local descriptors and their modifications [16-20], Gabor wavelets [18,21], etc. There are also many algorithms that combine these approaches like information fusion [22].

The method presented in this paper operates on facial features so it can be placed in the second group of algorithms. The partition of the input image into parts containing eyes, eyebrows, nose, and mouth can shorten the processing time and help avoid the problems of partially distorted images, e.g., poor lighting conditions, different facial expressions, the presence of glasses, beard, etc. Selected areas can be used for classification purposes. In partially distorted image, segments with good quality can be utilized. In addition, the classification based on the results of multiple classifiers leads to a number of results, which should then be aggregated to indicate at the end only one recognized person.

A. *Aggregation function*

An aggregation function is defined as $f:[0,1]^n \rightarrow [0,1]$ exhibiting the following properties [23]:

- bound preservation:

$$f(0, 0, \ldots, 0) = 0, \quad (1)$$

$$f(1,1, \ldots, 1) = 1, \quad (2)$$

- monotonicity:

$$\forall \boldsymbol{x}, \boldsymbol{y} \in [0,1]^n \; \mathbf{x} \leq \mathbf{y} \Rightarrow f(\mathbf{x}) \leq f(\mathbf{y}). \quad (3)$$

The vector inequality $\boldsymbol{x} \leq \boldsymbol{y}$ denotes that $x_i \leq y_i$ i=1, 2, …, n

For example, one of the classes of aggregation functions are the generic mathematical means. A number of aggregation operators can be recalled here: averaging, conjunctive, disjunctive, or mixed. They can also exhibit various properties including idempotency, symmetry, existence of neutral element, etc.

B. *Proposed method*

The analyzed face recognition process consists of the following steps. First, facial segments (regions) such as areas of eyebrows, eyes, nose and mouth are extracted from an image. Second, these elements are compared with parts of photos coming from the database and distances are determined. Third, the results are aggregated using various aggregation functions. The overall processing scheme is shown in Fig. 1. In this study, aggregation functions come in the form of some well-known utility functions. Utility functions are derived from the area of decision-making, where the decision-maker has to choose from a number of admissible solutions of a given problem. Each of these solutions can have many features

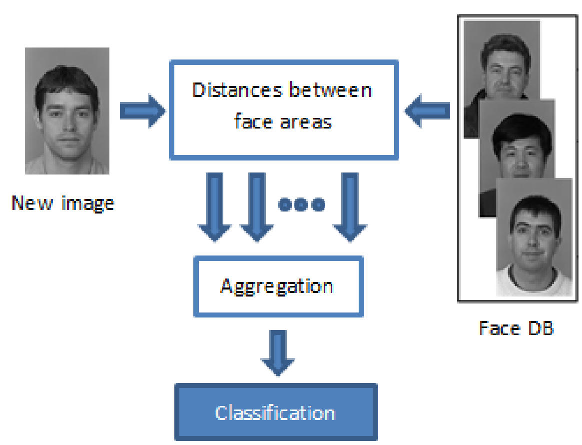

Fig. 1. The general processing scheme.

and aspects that must be taken into account. Finally, these solutions must be sorted, so it is useful to assign to each of them a number - value using which it will be sorted. Typically, the solution, which maximizes the utility function, becomes chosen.

Facial images are preprocessed including cropping, scaling, and histogram equalization. Next, the positions of salient facial regions are manually marked. Selected areas such as eyes, eyebrows, nose, and mouth are distinguished. For these four segments of images the well-known PCA and LDA methods are performed.

C. *Classifiers and aggregation functions*

The PCA and LDA results are compared by using the nearest neighbor classifier and involving various similarity/dissimilarity measures. This classifier is chosen because it is commonly used, effective, and intuitive. We use, inter alia classic Euclidean, Manhattan, Chebyshev, cosine, correlation, Bray-Curtis, Canberra, χ^2-statistics. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be two vectors $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, $\boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in \mathbb{R}^n$.The weighted form of the squared Euclidean is defined in the form

$$d(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\sqrt{w_i}}, \quad (4)$$

where w_i, $i = 1, \ldots, n$are the eigenvalues obtained using the PCA or LDA.

Let d_{eb}, d_{eo}, d_n, d_m denote distances between the vectors representing areas of the eyebrows, eyes, nose, and mouth of two persons A and B, respectively, and let d be a sequence of these distances, where $d_1=d_{eb}$, $d_2=d_{eo}$, *etc*. In addition, let $f(\bullet)$ be an aggregation function.

The calculated distances are then normalized to the interval [0,1] and aggregated using one of three, described below, aggregation functions in two variants each. The values of the considered function are calculated directly from the obtained distances, according to the scheme $f(d_{eb}, d_{eo}, d_n, d_m)$ or using associativity property like in following formula $f(f(f(d_{eb}, d_{eo}), d_n), d_m)$. Hence, there are considered six kinds of aggregation operators: f, g, h which are based on two factor utility function and are calculated directly from distances (first scheme) and F, G, H that are calculated sequentially (second scheme).

The first of analyzed functions is the formula, which is a result of the use of the two factors utility function by Kulikowski and attempting to match it to the well-known Hurwicz function. It has been used in the previous study [4]. In the series of experiments presented in this work we use the following expression

$$f(d,\alpha) = \left[min_{i=1,..,4}(d_i)\right]^{\alpha} \cdot \left[max_{i=1,..,4}(d_i)\right]^{1-\alpha}. \quad (5)$$

In the case of directly calculated aggregation function the sequence of distances has four elements (d_{eb}, d_{eo}, d_n, d_m) according to above-mentioned scheme.

The experiments are also carried out using associativity property of this function and will be denoted by uppercase function F. In this scenario, the function always takes only two arguments and it is calculated sequentially. Namely, it takes first and second distance, then the obtained result is placed into next calculation step as argument at the first position. The second argument is the next distance, and aggregation function value is calculated once more, and so on until all distances will be used. There will be also carried out analysis of other functions.. In this study, we consider $g(.)$ expressed in the following form

$$g(d,\alpha) = \left[min_{i=1,..,4}(d_i)\right]^{\alpha} + \left[max_{i=1,..,4}(d_i)\right]^{1-\alpha}, (6)$$

where the length of sequence analyzed in min and max is described in the same way as in previous function, but operator is changed. Once again the associative version of this function will be denoted by G. It also has been introduced an additional parameter, whose task is to increase the intervals between successive partial results, that in consequence influenced the final result of classification, namely

$$h(d,\alpha) = \left\{\left[min_{i=1,..,4}(d_i)\right]^{\alpha} \cdot \left[max_{i=1,..,4}(d_i)\right]^{1-\alpha}\right\}^{\beta}, (7)$$

where min and max are calculated as mentioned above. The associative function will be marked by uppercase H.

III. Experimental studies

The simulation was performed by using AT&T and FERET databases with the usage of PCA and LDA algorithms. AT&T (formerly Olivetti-Oracle Research Lab, ORL) database [7] contains 400 images of 40 people. Hence, for each person it is 10 photos taken with different lighting, pose, and facial expressions. The considered subset of Facial Recognition Technology (FERET) [8] contains 600 pictures of 200 people. Each person has three images taken once again with different facial expression and lighting. During the verification of the classification process, the AT&T database were divided by taking five randomly chosen images of one person to the training and five to the testing set, respectively. Similarly we conducted the series of tests for FERET database, namely two photos of one person were taken to the training and one to the testing set, respectively. Such procedure was repeated 200 times. Each time new images were chosen to experiments.

TABLE I. Average recognition rank (%) for three best norms.

Database	Method	Norm	Average recognition rank
AT&T	LDA	Cosine	86
		Correlation	86
		Bray-Curtis	85
	PCA	Manhattan	81
		Weighted squared Euclidean	81
		Bray-Curtis	80
FERET	LDA	Cosine	72
		Correlation	72
		Bray-Curtis	71
	PCA	Canberra	54
		Weighted Manhattan	37
		Weighted modified Manhattan	29

The aggregation function is calculated directly from distances (functions *f*, *g* and *h*) and associatively (functions *F*, *G*, and *H*). The distances used here were obtained using above-mentioned similarity/dissimilarity measures. The best result was obtained for classic Euclidean, Manhattan, cosine, correlation, Bray-Curtis, and weighted form of squared Euclidean measure.

A. Average recognition rank

Presented experiments produced very good classification results. According to them one can determinate the best similarity measures for each database and method. There were selected measures that allow us obtaining the best classification rank. Table 1 shows the best similarity functions of considered measures and average recognition ranks for all analyzed functions from all 200 iterations of the test. The results of PCA for FERET were not satisfactory, because the obtained recognition results were below 45% for almost all analyzed measures, only for Canaberra norm, the result was better, but still only about 60% in the best cases. This can be the result of a small number of images present in learning and testing sets. In the AT&T we have 5 photos per person in learning and 5 in testing set, when in FERET only 2 in learning and 1 in testing set are provided. LDA method leads to good results for both databases but slightly worse for FERET and PCA method produced good results for AT&T and low recognition rate for FERET.

TABLE II. AVERAGE RECOGNITION RANK (%) FOR SELECTED AGGREGATION FUNCTIONS

	Norm	Aggregation function					
		f	**F**	**g**	**G**	**h**	**H**
AT&T LDA	Cosine	87	90	87	89	79	84
	Correlation	87	90	87	89	79	84
	Bray-Curtis	87	89	86	88	78	83
FERET LDA	Cosine	70	79	70	79	60	74
	Correlation	70	79	69	79	59	73
	Bray-Curtis	69	78	68	78	57	73
AT&T PCA	Manhattan	83	84	83	84	73	79
	Weighted squared Euclidean	82	84	83	84	73	79
	Bray-Curtis	82	82	81	81	73	78
FERET PCA	Canberra	50	61	49	61	44	57
	Weighted Manhattan	34	43	32	40	32	40
	Weighted modified Manhattan	27	34	24	29	26	32

It is worth noting that for LDA method in both datasets we obtained the same set of the best norms. Also Bray-Curtis norm appears in all analyzed cases except FERET PCA, where the best obtained norms were different.

Subsequently they will be presented details for selected norms and analyzed aggregation functions. Table 2 shows average recognition results for functions *f, F, g, G, h and H* from all 200 test iterations. In all analyzed cases function calculated associatively yielded better results. And the first function *F* is better than others.

B. Analysis of parameters of aggregation functions

In the next step, the impact of parameters of aggregation functions is analyzed. The values which give the best recognition rank were determined using AT&T and FERET databases. Fig. 2 displays recognition rates obtained for functions *F* and *G* using the three best norms. Here AT&T database and LDA method were used.

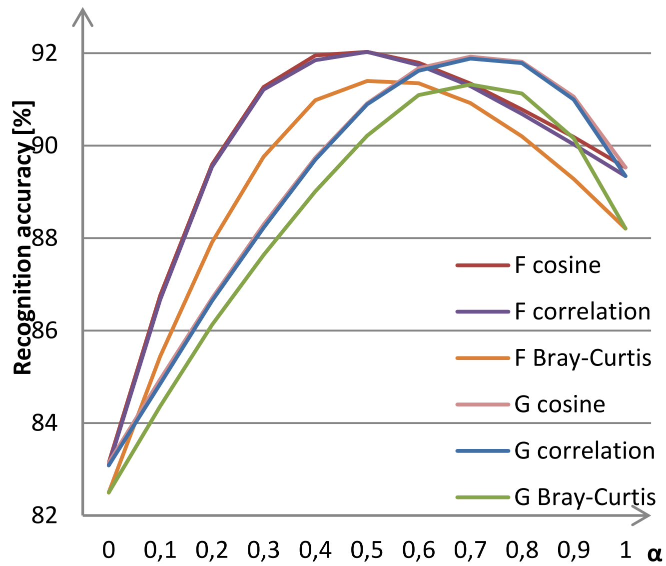

Fig. 2. Facial recognition rates (%) for function *F* and *G* (AT&T, LDA)

For α=0.5 function F with cosine and correlation norms allows to obtain 92% recognition rate. The same situation can be observed for α=0.7 and function *G* with the same norms. Bray-Curtis norm gives little lower results but also exceeds 90% for some values of α. Similar dependences were obtained for FERET database and LDA method (see Fig. 3).

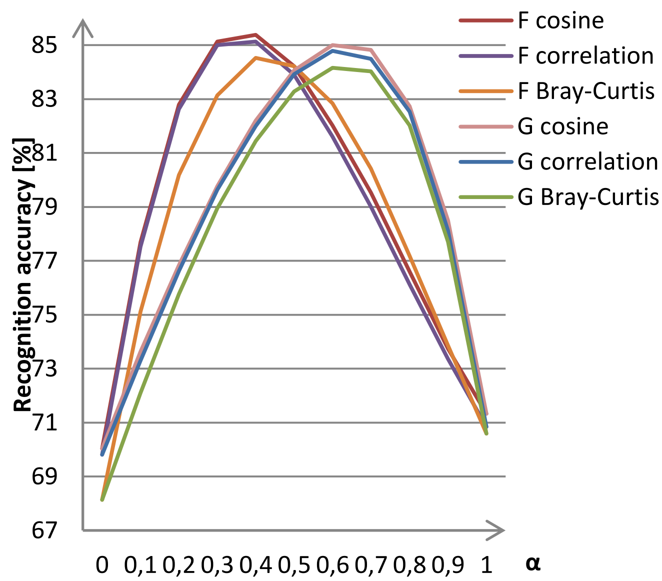

Fig. 3. Facial recognition rates (%) for function *F* and *G* (FERET, LDA)

For PCA method and AT&T database graph is more complicated (Fig. 4), but the above-described relationship is similar. Here we have also two norms which exceeds 86% for some values of α and Bray-Curtis norm with lower, but still good results.

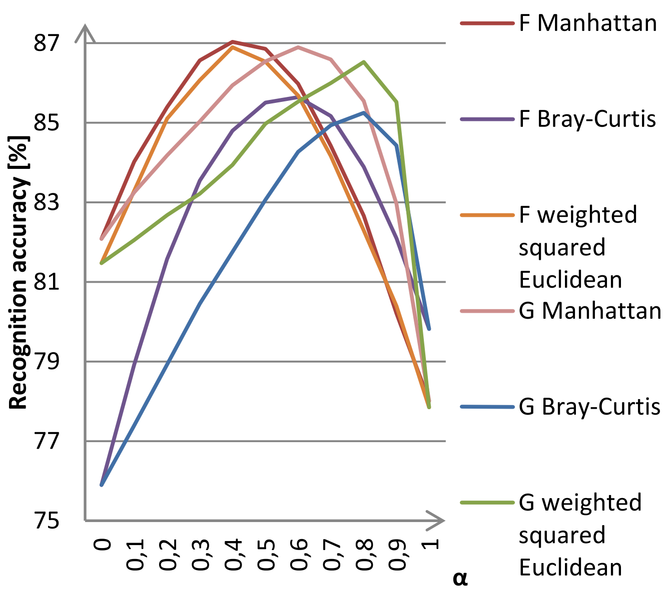

Fig. 4. Facial recognition rates (%) for functions *F* and *G* (AT&T, PCA)

For both considered databases LDA method allows to achieve better face recognition results than the PCA.

The third proposed function *h* and its associative version *H* depend on two parameters, so its visualization requires the three-dimensional graphs. In Fig. 5 there are presented typical plots obtained for one method (namely PCA) and aggregation function *h* in direct and associative version *H*. All described variants of the aggregation function lead to the plot of very similar shape, but different average quality of recognition.

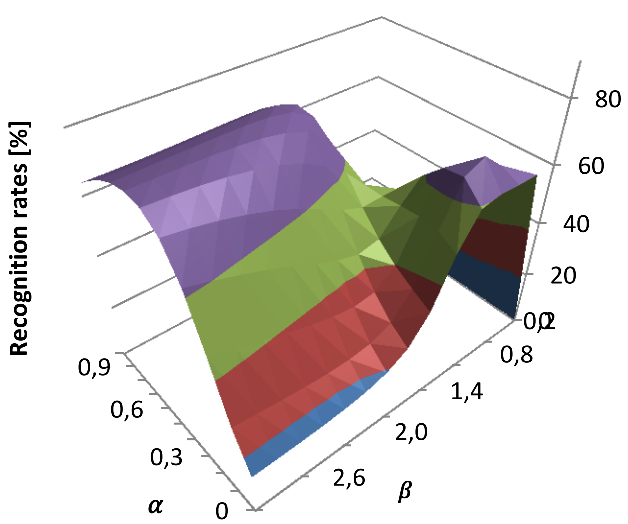

Fig. 5. Facial recognition rates (%) for function h, for FERET database, PCA

It is noteworthy that the presented here results with the aggregation of different classifiers are better than the results yielded for each classifier separately. The previous results for single classifiers can be found in [24].

Conclusions

The presented application of utility functions viewed as aggregation operators allows us to achieve better results of face recognition than when realizing classification based on only a single classifier (namely, based on the whole face). The parameters present in the aggregation functions offer an additional level of flexibility and adjust their forms to cope with the specificity of the dataset and the method being used for dimensionality reduction.

It is worth stressing that these aggregation functions were adopted from the theory of utility functions present in decision algorithms. This opens up interesting possibilities for carrying out further analysis at the junction of these research fields. We may anticipate that further studies may focus on studies of various aggregation functions (e.g., log utility) and their usefulness in the proposed setting. Some other classifiers, multimodal face recognition and other biometric methods can be also investigated as well and a comparative studies could be considered.

Acknowledgments

The authors are supported by National Science Centre, Poland (grant no. 2014/13/D/ST6/03244). Support from the Canada Research Chair (CRC) program and Natural Sciences and Engineering Research Council is gratefully acknowledged (W. Pedrycz).

References

[1] D.A. Forsyth and J. Ponce, “Computer vision: a modern approach,” Pearson, 2015.

[2] M. Weinmann, "Visual features—from early concepts to modern computer vision," in: G.M. Farinella, S. Battiato, and R. Cippola (eds.), Advanced topics in computer vision, London: Springer, pp. 1-34, 2013.

[3] C. Cobb and P. H. Douglas, "A theory of production," Am Econ Rev, vol. 18, pp. 139–165, 1928.

[4] R. Kulikowski, “A theory of motivation and satisfaction with application to deci-sion support,” Ann Oper Res, vol. 51, pp. 265-281, June 1994.

[5] R. Kulikowski, “Portfolio optimization: two factors-utility approach,” Control Cybern, vol. 27, pp. 429-446, 1998.

[6] M. Dolecki and R. Kozera, “Two factor utility function as an extension of Hurwicz criterion in game theory,” Computing in Science and Technology: Rzeszów, pp. 71-79, 2015.

[7] AT&T Laboratories Cambridge, “The Database of Faces,” Available: www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html [Accessed: Aug. 1, 2016].

[8] P. J. Phillips, J. Wechsler, J. Huang, and P. Rauss, “The FERET database and evaluation procedure for face recognition algorithms,” Image and Vision Computing, vol. 16, no. 5, pp. 295–306, April 1998.

[9] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” ACM Comput Surv, vol. 35, pp. 399-458, December 2003.

[10] M. Turk and A. Pentland, “Eigenfaces for recognition,” J Cogn Neurosci, vol. 3, pp. 71-86, 1991.

[11] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman “Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection,” IEEE Trans Pattern Anal Mach Intell, vol. 19, no. 7, pp. 711-720, July 1997.

[12] P. J. Phillips, “Support vector machines applied to face recognition,” Proc. 1998 Conference on Adv Neural Inf Process Syst vol. 11, pp. 803–809, 1998.

[13] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “DeepFace: closing the gap to human-level performance in face verrification,” Proc. of CVPR 2014, pp. 1701-1708, 2014.

[14] L. Wiskott, J. M. Fellous, N. Krüger, and C. von der Malsburg, “Face recognition by elastic bunch graph matching,” IEEE Trans Pattern Anal Mach Intell, vol. 19, pp. 775-779, July 1997.

[15] T. Kanade, Computer recognition of human faces, Basel: Birkhauser Verlag, 1977.

[16] T. Ahonen, A. Hadid, and M. Pietikäinen, “Face recognition with local binary patterns,” Proc. of the 8th European Conference on Computer Vision LNCS, vol. 3021, pp. 469-481, 2004.

[17] M. Heikkilä, M. Pietikäinen, and C. Schmid “Description of interest regions with local binary patterns,” Pattern Recognit, vol. 42, pp. 425-436, March 2009.

[18] M. Bereta, P. Karczmarek, W. Pedrycz, and M. Reformat, “Local descriptors in application to the aging problem in face recognition,” Pattern Recognit, vol. 46, no. 10, pp. 2634-2646, October 2013.

[19] M. Bereta, W. Pedrycz, and M. Reformat, “Local descriptors and similarity measures for frontal face recognition: A comparative analysis,” J Vis Commun Image Rep, vol. 24, no. 8, pp. 1213–1231, November 2013.

[20] F. Dornaika, A. Bosaghzadeh, H. Salmane, and Y. Ruichek, “Graph-based semi-supervised learning with Local Binary Patterns for holistic object categorization”, Expert Syst Appl, vol. 41, no. 17, pp. 7744–7753, 2014.

[21] S. Xie, S. Shan, X. Chen, and J. Chen, “Fusing local patterns of Gabor magnitude and phase for face recognition”, IEEE Transactions on Image Processing, vol. 19, no. 5, pp. 1349–1361, May 2010.

[22] K.-C. Kwak and W. Pedrycz, “Face recognition: a study in information fusion using fuzzy integral,” Pattern Recognit Lett, vol. 26, no. 6, pp. 719–733, May 2005.

[23] G. Beliakov, A. Pradera, and T. Calvo, Aggregation functions: A guide for practitioners, Berlin: Springer, 2008.

[24] P. Karczmarek, W. Pedrycz, M. Reformat, and E. Akhoundi, “A study in facial regions saliency: A fuzzy measure approach,” Soft Comput, vol. 18, no. 2, pp. 379-391, February 2014.