

2D face recognition using PCA and triplet similarity embedding

Bek Bazatbekov¹, Cemil Turan¹, Shirali Kadyrov², Askhat Aitimov¹

¹Department of Computer Science, Faculty of Engineering and Natural Science, Suleyman Demirel University, Kaskelen, Kazakhstan

²Department of Mathematics, Faculty of Engineering and Natural Science, Suleyman Demirel University, Kaskelen, Kazakhstan

Article Info

Article history:

Received May 29, 2022

Revised Oct 7, 2022

Accepted Oct 26, 2022

Keywords:

Face recognition

K-nearest neighbor

Principal component analysis

TP

Stochastic gradient descent

Triplet loss

ABSTRACT

The aim of this study is to propose a new robust face recognition algorithm by combining principal component analysis (PCA), Triplet Similarity Embedding based technique and Projection as a similarity metric at the different stages of the recognition processes. The main idea is to use PCA for feature extraction and dimensionality reduction, then train the triplet similarity embedding to accommodate changes in the facial poses, and finally use orthogonal projection as a similarity metric for classification. We use the open source ORL dataset to conduct the experiments to find the recognition rates of the proposed algorithm and compare them to the performance of one of the very well-known machine learning algorithms k-Nearest Neighbor classifier. Our experimental results show that the proposed model outperforms the kNN. Moreover, when the training set is smaller than the test set, the performance contribution of triplet similarity embedding during the learning phase becomes more visible compared to without it.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Askhat Aitimov

Department of Computer Science, Faculty of Engineering and Natural Science

Suleyman Demirel University

Kaskelen, Kazakhstan

Email: askhat.aitimov@sdu.edu.kz

1. INTRODUCTION

Pattern recognition and computer vision are the fast-developing fields of computer science. Face recognition is the principal biometric method and compared to other methods such as fingerprint and iris recognition it does not require cooperation with the device. Face recognition finds applications in forensic investigations, identity verification in security points, healthcare, and game apps. Despite the promising progress in the field there are still many challenges due to the dynamic nature of the task including illumination, pose variations, occlusion, aging, and facial expressions [1]. Various techniques and algorithms were proposed to better identify and recognize facial images [2] and we briefly review some of the well-studied and classical approaches for still-image recognition. From a purely mathematical perspective, one can call a classifier linear or non-linear based on whether the techniques underpinning the algorithm make use of linear algebra or (differential) calculus, respectively. We first start recalling some linear techniques.

One of the classical approaches is Eigenfaces [3], [4] which is based on principal component analysis (PCA). PCA is by far the most well-known unsupervised dimension reduction and the feature extraction method used in many disciplines. The main idea is to find an approximate basis consisting of eigenvectors of a covariance matrix from images in the dataset. For a detailed algorithm, we refer to the next section.

Another closely related technique is fisherfaces based on linear discriminant analysis (LDA) [5], [6]. This is a class-specific approach where one tries to better arrange images before PCA is applied. The main distinction between PCA and LDA is that the latter tries to better understand the distinctions between classes of faces. k-Nearest Neighborhood (kNN) classifier is a common supervised machine learning method used in

clustering problems. Being simple, fast and having a high recognition rate make the kNN algorithm a popular classifier [7]. For a given positive integer k and the space of labeled images, the kNN finds k -nearest neighbors of an image to be classified according to a distance function. The image then gets labeled according to the majority votes of these k -nearest neighbors.

All three classifiers mentioned above are examples of linear models. The last linear model we review here is the sparse representation based classifier (SRC) [8], [9]. In this model, a new face image is written as a linear combination of labeled images in the training set with l_1 -norm minimization and it is labeled according to the class that receives the maximum weight. While linear models receive high recognition rates for various available databases, the state of the art in face recognition techniques for large datasets is the family of non-linear models based on deep learning methodology [10]-[12]. More specifically, convolutional neural network (CNN) based approaches to face detection and recognition receive many real-life applications. In simple terms, the main idea of deep learning is to train a nonlinear face recognition function using the stochastic gradient descent (SGD) algorithm.

Various works consider improvements of the above-mentioned algorithms and compare the recognition rate to others using certain databases of faces. The performance of PCA with nearest distance criteria was compared against the LDA with Euclidean distance criteria in [13] using AT&T (ORL) database [14] and LDA showed better performance with a maximum recognition rate of 95% when 90% of the database images were allocated to the training set. Another work [15] also uses an ORL database to compare the accuracy of the original SRC to the proposed discriminative common vector dictionary-based SRC (DCV-SRC). The maximum accuracy of 96.54 ± 1.69 is reached with DCV-SRC when 70% of the ORL dataset is split into the training set while SRC showed an accuracy of 95.58 ± 2.24 . In a recent work [16], a version of the kNN classifier based on kernel discriminant analysis together with a support vector machine was proposed and it was reported to have 96% of recognition accuracy when the ORL database is used with 60% split into the training set. Lastly, we note that when 80% of the ORL faces were allocated to the training set, the CNN reported having a recognition rate of 95% [17].

Our goal in this article is to propose an improvement to Eigenfaces and compare the novel algorithm to the above-mentioned algorithms using the ORL dataset of facial images. In the next section we introduce ORL dataset and the proposed algorithm in detail. Section three contains the results of the experiment. We end with a discussion and suggestions for future work.

2. METHOD

In this section, we provide information on the ORL dataset we study and the proposed (PCA-TP) classifier which merges PCA and triplet similarity embedding together with projection-based recognition metric. In the first part we use the principal component analysis for dimensionality reduction followed by triplet similarity embedding. The final part makes use of the orthogonal projection of a vector onto a subspace to predict the label.

2.1. ORL dataset

In this work, we use the ORL face database [18] which contains 400 images of 40 individuals (10 images per person) under different conditions such as lighting, expressions and taking time as shown in Figure 1 for a few samples and in Figure 2 for one sample. Images are grayscale and have a size of 112×92 . All images are split into training and test sets in a stratified manner, which means sets have the same proportion of subjects.



Figure 1. Samples from ORL dataset



Figure 2. 10 different images of one individual from ORL dataset

2.2. Proposed face classifier PCA-TP

We let $Train$ denote the labeled set $\{x_1, x_2, \dots, x_N\}$ of m dimensional vectors that will be used to train the classifier, $Test$ denote the labeled set $\{y_1, y_2, \dots, y_M\}$ of m dimensional vectors that will be used to test the accuracy of the classifier. We now introduce our proposal classifier PCA-TP. It has three essential parts, namely, dimension reduction with PCA, training of the triplet similarity embedding, and using the Projection as a similarity metric. We now explain each part in detail.

One of the well-known dimension reduction techniques is the principal component analysis (PCA) [19]. In computer vision, it is used to represent an image with a relatively small dimensional feature vector as we introduce: Let $A = [x_1 \ x_2 \ \dots \ x_n]$ be the $m \times n$ matrix with m -dimensional vectors from Train stacked into columns. We let μ be the row average of A , that is, the m -dimensional column vector $\mu = \frac{1}{n} \sum_{i=1}^n x_i$ and set $X_0 = X - \mu$ where the vector μ is subtracted from each column of X . Next, we consider the covariance matrix $\frac{1}{n-1} X_0 X_0^T$ which is readily seen to be symmetric. Hence, its eigenvalues are all real and we may list them in decreasing order $\{\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n\}$. Let $\{e_1, e_2, \dots, e_n\}$ be the corresponding eigenvectors. To reduce the dimension m into d , we form a $d \times m$ matrix X_{pca} from the first d eigenvectors $\{e_1, e_2, \dots, e_d\}$. For our purposes, we simply take d to be the number of nonzero eigenvalues counting the multiplicity. As the nonzero eigenvalues of $X_0 X_0^T$ and $X_0^T X_0$ are identical, we see that d is at most (m, n) . Once X_{pca} is computed, any m -dimensional vectors x representing an image can be transformed into a d -dimensional feature vector x' via $x' = X_{pca} (x - \mu)$.

2.3. Triplet similarity embedding

Triplet similarity embedding is a deep learning method [20] used in computer vision to improve face recognition, verification and clustering. Idea was to train an embedding f from the space of feature vectors so that for any anchor $\{a\}$ and positive $\{p\}$ with the same label and a negative $\{n\}$ with a different label, it should hold $dist(f(a), f(p)) < dist(f(a), f(n))$ for a given metric $dist(,)$. To avoid trivial solutions a positive hyperparameter $\alpha > 0$ is included and then the training set is used to minimize the (1):

$$\max(0, \alpha + dist(f(a), f(p)) - dist(f(a), f(n))) \text{ for all triplets } \{a, p, n\} \subset Train \quad (1)$$

In this article, a linear transformation f is trained and projection (dot product) based approach adopted from [21] is used. The formula $\cos(\theta) = (u, v)$, where $(,)$ is the dot product operation, gives the cosine of the angle θ between two unit vectors u, v . Therefore, if u, v are feature vectors representing the same face, they are expected to be close to one other, making θ close to zero and thus $\cos(\theta)$ large. Geometrically, if v has a unit length, the dot product till the sign can be assumed as an Euclidean norm of projection of u onto the span of v . The closer vectors make the dot product larger in the projection setting and then one needs to train the similarity transformation f so that (2) is minimal.

$$\max(0, \alpha + (f(a), f(n)) - (f(a), f(p))) \quad (2)$$

For a linear transformation f , we consider $f(x) = Wx$ for some $r \times d$ matrix W where d is the dimension of feature vectors obtained as a result of dimension reduction due to PCA and integer $r \leq d$ is a hyperparameter. Hence, W reduces the dimension feature vectors from d further to r . Note that $(f(x), f(y)) = f(x)^T f(y)$. Hence, the minimization problem has the final form (3):

$$\operatorname{argmin}_W \sum_{\{a,p,n\}} \max(0, \alpha + a^T W^T W n - a^T W^T W p), \quad (3)$$

where $\{a, p, n\}$ runs from the training set of feature vectors. To optimize the matrix W , the stochastic gradient descent (SGD) is used together with l_2 -regularization to avoid overfitting. As the gradient of an individual loss satisfies (4):

$$\nabla(a^T W^T W n - a^T W^T W p) = W(an^T + na^T) - W(ap^T + pa^T), \quad (4)$$

in each iteration of SGD the W gets updated (5) as:

$$W_{t+1} = W_t - \beta W(an^T + na^T) + (n - p)a^T - \gamma W \quad (5)$$

where $\gamma \mathbf{W}$ is the l_2 -regularization. For the experiment, we iterate SGD 1,000,000 times for the hyperparameters $\alpha = \beta = 0.001$, $\gamma = 0.00001$ and initiate our matrix \mathbf{W} as a $d \times d$ identity matrix.

2.4. Projection as a similarity metric

In computer vision there are various metrics considered to test if two vectors belong to the same label, see e.g [22] in which 15 different distance functions are used. We just use the Euclidean norm of the projection onto a subspace. The orthogonal projection for an $m \times m$ matrix onto the subspace V is defined as $P = A(A^T A)^{-1} A^T$ (see [23]) where A is an $m \times n$ matrix with columns consisting of n basis vectors of V . It is readily seen that P satisfies $P^T = P$ and $P^2 = P$. In our setting, V is a subspace that holds different images for a single label. Images from the test set may be assumed to share the same label if their Euclidean norm of $P\mathbf{y}$ for a feature vector \mathbf{y} is higher than projections onto subspaces of other labels. The Euclidean norm square for $P\mathbf{y}$ satisfies (6):

$$\|P\mathbf{y}\|_2^2 = (P\mathbf{y}, P\mathbf{y}) = \mathbf{y}^T P^T P \mathbf{y} = \mathbf{y}^T P^2 \mathbf{y} = \mathbf{y}^T P \mathbf{y} \quad (6)$$

Therefore, this formula can be used to compare the projection norms for each labeled class. In order to label a test feature vector \mathbf{y} , the following can be written as (7):

$$\mathbf{y} \text{ label} = \operatorname{argmax}_i \mathbf{y}^T \mathbf{P}_i \mathbf{y} \quad (7)$$

where \mathbf{P}_i is the matrix that is projected to the subspace formed by the feature vectors from the training set that belong to the label i . We point out that \mathbf{A} has to include the columns that are linearly independent, and particularly the training set images need not to be redundant. To summarize we have the following short description of Algorithm 1. We denote PCA-P as the same algorithm except the triplet similarity embedding is ignored, that is, SGD is not implemented.

Algorithm 1. Proposed PCA-TP algorithm

Input: $m \times n$ matrix $\mathbf{A} = [\mathbf{A}_1 \mathbf{A}_2 \dots \mathbf{A}_n]$ for n labels, $\mathbf{y} \in R^m$, $r, t \in N$, $\alpha, \beta, \gamma > 0$, $\omega \in [0, 1]$

Output: label for \mathbf{y}

Columns of \mathbf{A} and \mathbf{y} are normalized w.r.t. l_2 -norm

PCA is implemented to reduce \mathbf{A} to an $d \times n$ matrix \mathbf{A}'

\mathbf{y} is transformed into \mathbf{y}' in R^d with PCA transformation

Set $\mathbf{W} = r \times d$ as identity matrix

Train a triplet similarity with SGD for $k \times d$ matrix \mathbf{W} according to (1):

for $i = 1$ to t do

 take random $\{\mathbf{a}, \mathbf{p}, \mathbf{n}\}$ from columns \mathbf{A} with $\{\mathbf{a}, \mathbf{p}\}$ the same label.

 if $\alpha + \mathbf{a}^T \mathbf{W}^T \mathbf{W} \mathbf{n} - \mathbf{a}^T \mathbf{W}^T \mathbf{W} \mathbf{p} > 0$ do

$\mathbf{W} := \mathbf{W} - \beta \mathbf{W} (\mathbf{a}(\mathbf{n} - \mathbf{p})^T + (\mathbf{n} - \mathbf{p})\mathbf{a}^T) - \gamma \mathbf{W}$

 end for

 Compute $\mathbf{A}'' = \mathbf{W} \mathbf{A}'$ and $\mathbf{y}'' = \mathbf{W} \mathbf{y}'$

 for $i = 1$ to n do

 Calculate the similarity metric R_i regarding to (2)

 end for

 Assign label = $\operatorname{arg arg} \{R_i\}$

3. RESULTS AND DISCUSSION

This work compares the performance of kNN to the proposed algorithm PCA-TP using the ORL dataset. Four different scenarios were considered where the dataset is randomly split into training and test sets with ratios 80/20, 60/40, 40/60, and 20/80. We compare the accuracy of our finding to that of the k-Nearest Neighborhood classifier (kNN). We only report kNN accuracy for $k = 1$ as the greater values of k underperforms. The results of our experiments are summarized in Table 1.

Table 1 shows that both PCA-TP and PCA-P have better accuracy almost for all types of train/test portions than that of KNN (for detailed information on kNN we refer to [24] and [25]). It is clearly seen that PCA-TP has superiority over other algorithms for all conditions. We note that with the increase of the training set, the similarity embedding methodology has not really improved the classifier. On the other hand, for instance, when the training set has as few as 20% of the dataset, additional 3% accuracy is supplied by triplet similarity embedding.

The graphs of loss function f (similarity embedding transformation) from the training of the similarity embedding with SGD for each train/test ratio respectively from high to low are given in Figure 3. The graphs show that there exists more distortion for a low ratio of train/test split and as seen in Table 1, this distortion influences the accuracy of the algorithm. Table 2 provides some of the commonly mislabeled images and their false positives. Here false matching is understandable because the pairs are almost similar to each other.

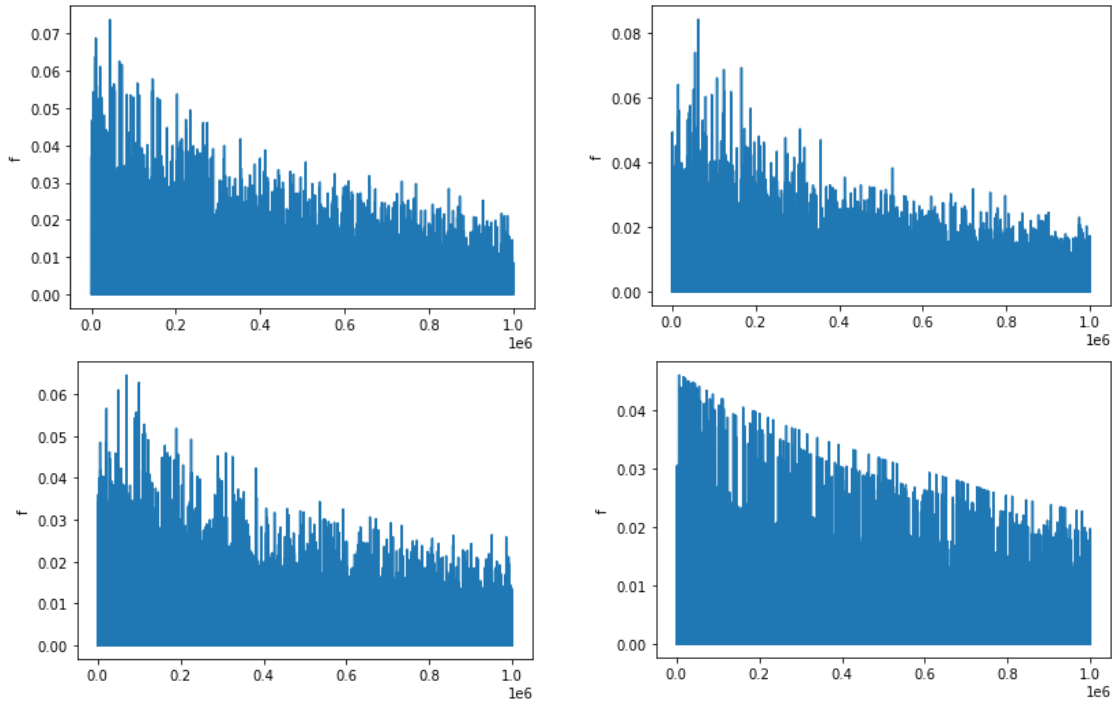
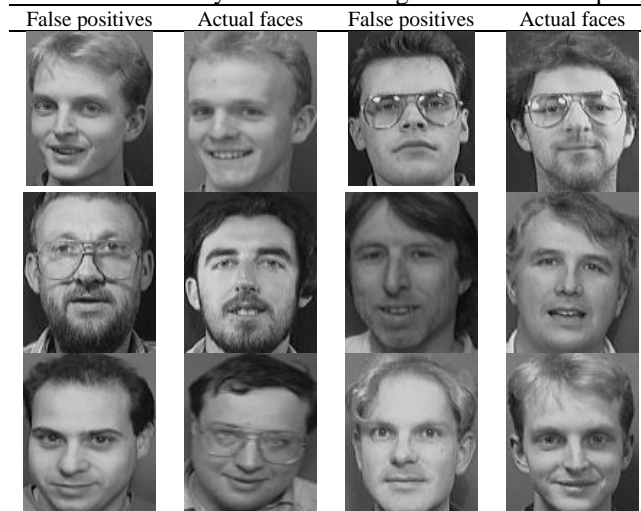


Figure 3. Training triplet similarity embedding matrix with SGD for various splitting of the dataset (from 80/20 to 20/80 train/test ratio)

Table 1. Accuracy of classifiers

Train/test split (%)	PCA-TP (%)	PCA-P (%)	kNN (k=1) (%)
80/20	98.75	98.75	95
60/40	97.5	97.5	95
40/60	93.33	92.5	92.5
20/80	84.375	81.56	81.88

Table 2. The commonly mislabeled images and their false positives






4. CONCLUSION

In this work, we proposed a new robust algorithm for face recognition. It was established by combining the well-known principal component analysis to get a dimensionality reduction first. One of the effective deep learning methods, triplet similarity embedding was applied secondly for the learning process and finally used the Projection as a similarity metric for classification. The proposed algorithm was applied to ORL dataset for different train /test portions and the results were compared to that of the PCA-P ,that is without triplet similarity, to check the effect of it and then to one of the well-known machine learning algorithms kNN. The comparisons showed that the combination of the three methods which gives us PCA-TP is a well-established algorithm for higher performance. It has superiority over others almost for all conditions. Unfortunately, there exists a high deterioration of the recognition rate with low train /test value. As a future work one can develop an algorithm to compensate for this rapid decrease in accuracy for lower train sets.




REFERENCES

- [1] M. Hassaballah and S. Aly, "Face recognition: Challenges, achievements and future directions," *IET Computer Vision*, vol. 9, no. 4, pp. 614–626, 2015, doi: 10.1049/iet-cvi.2014.0084.
- [2] Z. Mahmood, N. Muhammad, N. Bibi, and T. Ali, "A review on state-of-the-art face recognition approaches," *Fractals*, vol. 25, no. 2, pp. 1–19, Apr. 2017, doi: 10.1142/S0218348X17500256.
- [3] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 103–108, 1990, doi: 10.1109/34.41390.
- [4] A. L. Machidon, O. M. Machidon, and P. L. Ogrutan, "Face recognition using eigenfaces, geometrical PCA approximation and neural networks," in *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, Jul. 2019, pp. 80–83, doi: 10.1109/TSP.2019.8768864.
- [5] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, Jul. 1997, doi: 10.1109/34.598228.
- [6] I. Gangopadhyay, A. Chatterjee, and I. Das, "Face detection and expression recognition using Haar cascade classifier and Fisherface algorithm," in *Advances in Intelligent Systems and Computing*, 2019, pp. 1–11, doi: 10.1007/978-981-13-6783-0_1.
- [7] K. Sasirekha and K. Thangavel, "Optimization of K-nearest neighbor using particle swarm optimization for face recognition," *Neural Computing and Applications*, vol. 31, no. 11, pp. 7935–7944, Nov. 2019, doi: 10.1007/s00521-018-3624-9.
- [8] J. Wright, A. Ganesh, A. Yang, and Y. Ma, "Robust face recognition via sparse representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 2, pp. 210–227, 2007.
- [9] C. Turan and R. Jantayev, "Sparse representation based face recognition under varying illumination," in *2018 14th International Conference on Electronics Computer and Computation (ICECCO)*, Nov. 2018, pp. 206–212, doi: 10.1109/ICECCO.2018.8634755.
- [10] S. Balaban, "Deep learning and face recognition: The state of the art," in *Biometric and Surveillance Technology for Human and Activity Identification XII*, May 2015, pp. 68–75, doi: 10.1117/12.2181526.
- [11] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1701–1708, doi: 10.1109/CVPR.2014.220.
- [12] M. Winter and W. Bailer, "Incremental training for face recognition," in *International Conference on Multimedia Modeling*, 2019, pp. 289–299, doi: 10.1007/978-3-030-05710-7_24.
- [13] M. Saraswathi and S. Sivakumari, "Evaluation of PCA and LDA techniques for face recognition using ORL face database," *International Journal of Computer Science and Information Technologies*, vol. 6, no. 1, pp. 810–813, 2015.
- [14] F. S. Samaria and A. C. Harter, "The ORL database of faces," *AT&T Laboratories Cambridge*, 1994. <http://cam-orl.co.uk/facedatabase.html> (Accessed 15.01.2022).
- [15] Y. Wen, "A novel dictionary based SRC for face recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, pp. 2582–2586, doi: 10.1109/ICASSP.2017.7952623.
- [16] M. Z. Al-Dabagh, M. H. M. Alhabib, and F. H. AL-Mukhtar, "Face recognition system based on kernel discriminant analysis, k-nearest neighbor and support vector machine," *International Journal of Research and Engineering*, vol. 5, no. 2, pp. 335–338, 2018, doi: 10.21276/ijre.2018.5.3.3.
- [17] H. Hu, S. A. A. Shah, M. Bennamoun, and M. Molton, "2D and 3D face recognition using convolutional neural network," in *TENCON 2017 - 2017 IEEE Region 10 Conference*, Nov. 2017, pp. 133–132, doi: 10.1109/TENCON.2017.8227850.
- [18] A. Chater and A. Lasfar, "New approach to the identification of the easy expression recognition system by robust techniques (SIFT, PCA-SIFT, ASIFT and SURF)," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 695–704, Apr. 2020, doi: 10.12928/telkomnika.v18i2.13726.
- [19] A. M. Alkababji and S. R. Abd, "Half-face based recognition using principal component analysis," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 3, pp. 1404–1410, Jun. 2021, doi: 10.11591/ijeecs.v22.i3.pp1404-1410.
- [20] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 815–823, doi: 10.1109/CVPR.2015.7298682.
- [21] S. Sankaranarayanan, A. Alavi, and R. Chellappa, "Triplet similarity embedding for face verification," Feb. 2016, [Online]. Available: <http://arxiv.org/abs/1602.03418>
- [22] V. Perlibakas, "Distance measures for PCA-based face recognition," *Pattern Recognition Letters*, vol. 25, no. 6, pp. 711–724, Apr. 2004, doi: 10.1016/j.patrec.2004.01.011.
- [23] D. Margalit and J. Rabinoff, *Interactive linear algebra*. Boston: Free Software Foundation, 2019.
- [24] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967, doi: 10.1109/TIT.1967.1053964.
- [25] A. Kataria and M. D. Singh, "A review of data classification using k-nearest neighbor algorithm," *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 6, pp. 354–360, 2013.




BIOGRAPHIES OF AUTHORS

Bek Bazatbekov    received Bachelor of Mathematics in SDU (Suleyman Demirel University) in 2010, Master degree also in SDU from Faculty of Engineering and Natural Science in 2019. Currently He is PhD student in Computer Science and jointly lecturer in SDU. His research interests include object recognition and building information modeling. He can be contacted at email: bek.bazatbekov@sdu.edu.kz.






Cemil Turan    is an Associated Professor at Suleyman Demirel University, Kazakhstan, where he has been a faculty member since 2008. From 2021-2022, he was also the Head of Computer Science department. Cemil graduated bachelor degree in Electrical Engineering from Yildiz Technical University, Turkey, in 1995, and completed his Ph.D. in Electrical and Computer Engineering from Mevlana Rumi University, Turkey, in 2016. Her research interests are primarily in the area of Digital Signal and Image Processing. He can be contacted at email: cemil.turan@sdu.edu.kz.



Shirali Kadyrov    is a Professor at Suleyman Demirel University, Kazakhstan, where he has been a faculty member since 2017. From 2018-2019, he was also the Head of the Science department. Shirali graduated bachelor degree in Mathematics from Bogazici University, Turkey, in 2004, and completed his Ph.D. in Mathematics from the Ohio State University, the USA, in 2010. Her research interests are primarily in the area of data science, dynamical systems, ergodic theorem and number theory. He can be contacted at email: shirali.kadyrov@sdu.edu.kz.



Askhat Aitimov    received Bachelor of Information system in IT University in 2009, Master degree in SDU from Faculty of Engineering and Natural Science in 2015. Currently He is PhD student in Computer Science, jointly He is the Vice dean in Faculty of Engineering and Natural Science and senior lecturer in SDU. His research interests include object and speech recognition. He can be contacted at email: askhat.aitimov@sdu.edu.kz.