RPGE Revista on line de Política e Gestão Educacional
unesp

BIOMETRIC TECHNOLOGY OF PERSONAL RECOGNITION

TECNOLOGIA BIOMÉTRICA DE RECONHECIMENTO PESSOAL

TECNOLOGÍA BIOMÉTRICA DE RECONOCIMIENTO PERSONAL

Alla KAPITON[1]
e-mail: kits_seminar@ukr.net
Nataliia KONONETS[2]
e-mail: natalkapoltava7476@gmail.com
Volodymyr MOKLIAK[3]
e-mail: vovchik01071981@gmail.com
Valentyna ONIPKO[4]
e-mail: valentyna.onipko@pdau.edu.ua
Serhiy DUDKO[5]
e-mail: dudko@pano.pl.ua
Vadym PYLYPENKO[6]
e-mail: pylypenko@pano.pl.ua
Anna SOKIL[7]
e-mail: sokol7227@ukr.net

How to reference this paper:

KAPITON, A.; KONONETS, N.; MOKLIAK, V.; ONIPKO, V.; DUDKO, S.; PYLYPENKO, V.; SOKIL, A. Biometric technology of personal recognition. **Revista on line de Política e Gestão Educacional**, Araraquara, v. 28, n. 00, e023015, 2024. e-ISSN: 1519-9029. DOI: https://doi.org/10.22633/rpge.v28i00.19390

| **Submitted**: 11/03/2024
| **Revisions required**: 08/04/2024
| **Approved**: 16/05/2024
| **Published**: 19/06/2024

Editor: Prof. Dr. Sebastião de Souza Lemes
Deputy Executive Editor: Prof. Dr. José Anderson Santos Cruz

[1] National University «Yuri Kondratyuk Poltava Polytechnic», Poltava – Ukraine. Associate Professor of the Department of Computer and Information Technologies and Systems. Doctor of Pedagogical Sciences.
[2] University of Ucoopspilka «Poltava University of Economics and Trade», Poltava – Ukraine. Associate Professor of the Department of Economic Cybernetics, Business Economics and Information Systems. Doctor of Pedagogical Sciences.
[3] Poltava V. G. Korolenko National Pedagogical University, Poltava – Ukraine. Professor of the Department of General Pedagogy and Andragogy. Doctor of Pedagogical Sciences.
[4] Poltava State Agrarian University, Poltava – Ukraine. Professor of the Department of Agriculture and Agrochemistry named after V. I. Sazanov, Department of the Construction and Professional Education. Doctor of Pedagogical Sciences.
[5] Poltava M. V. Ostrogradsky Academy of Continuous Education, Poltava – Ukraine. Deputy Director. PhD in Pedagogical Sciences.
[6] Poltava M. V. Ostrogradsky Academy of Continuous Education, Poltava – Ukraine. First Deputy Director. PhD of Pedagogical Sciences.
[7] Poltava V. G. Korolenko National Pedagogical University, Poltava – Ukraine. Postgraduate of the Department of General Pedagogy and Andragogy.

ABSTRACT: Today, image processing is widely used in security systems to recognize people. For this study, machine learning algorithms were selected, in the presence of a limited amount of data. The work analyzed the subject area of face recognition, the relevance of this system in our time, biometric recognition of the «FaceID» system, several different methods for face recognition, and investigated in which areas of activity the «Face recognition» system is used for which purpose. As part of the study, a number of face recognition algorithms were analyzed. It has been proven that the entire face recognition system can be modeled using the Violy-Jones contour extraction method and tested with a successful recognition result of approximately 75%. The functional capabilities of the OpenCV computer vision library and other libraries are considered.

KEYWORDS: Object recognition system. Algorithm. Computer vision. Biometric technologies.

RESUMO*: Atualmente, o processamento de imagens é amplamente utilizado em sistemas de segurança para reconhecer pessoas. Para este estudo, foram selecionados algoritmos de aprendizado de máquina, considerando a disponibilidade limitada de dados. O trabalho analisou a área de reconhecimento facial, a relevância deste sistema nos dias de hoje, o reconhecimento biométrico do sistema «Face ID», diversos métodos para o reconhecimento facial e investigou em quais áreas de atividade o sistema de reconhecimento facial é utilizado e para qual finalidade. Como parte do estudo, diversos algoritmos de reconhecimento facial foram analisados. Foi comprovado que todo o sistema de reconhecimento facial pode ser modelado utilizando o método de extração de contornos Violy-Jones e testado com um resultado bem-sucedido de reconhecimento de aproximadamente 75%. As capacidades funcionais da biblioteca de visão computacional OpenCV e outras bibliotecas foram consideradas.*

PALAVRAS-CHAVE*: Sistema de reconhecimento de objetos. Algoritmo. Visão computacional. Tecnologias biométricas.*

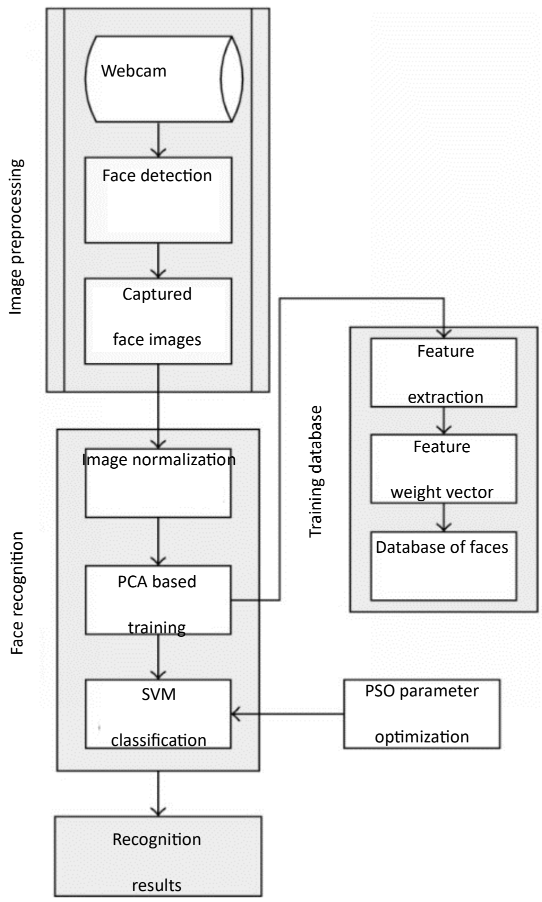
RESUMEN*: En la actualidad, el procesamiento de imágenes se utiliza ampliamente en los sistemas de seguridad para reconocer personas. Para este estudio se seleccionaron algoritmos de aprendizaje automático, en presencia de una cantidad limitada de datos. El trabajo analizó el área temática del reconocimiento facial, la relevancia de este sistema en nuestro tiempo, el reconocimiento biométrico del sistema "FaceID", varios métodos diferentes para el reconocimiento facial, e investigó en qué áreas de actividad se utiliza el sistema "Face recognition" y con qué propósito. Como parte del estudio, se analizaron varios algoritmos de reconocimiento facial. Se ha comprobado que todo el sistema de reconocimiento de caras puede modelarse utilizando el método de extracción de contornos Violy-Jones y se ha probado con un resultado de reconocimiento satisfactorio de aproximadamente el 75%. Se consideran las capacidades funcionales de la biblioteca de visión por ordenador OpenCV y de otras bibliotecas.*

PALABRAS CLAVE*: Sistema de reconocimiento de objetos. Algoritmo. Visión por computadora. Tecnologías biométricas.*

Introduction

Let's consider the main concepts of the face recognition system. A technology capable of detecting and recognizing one or more persons in an image or sequence of video frames is a face recognition system. In such systems, various identification methods exist for comparing a given image with a face in the database. Facial recognition systems are increasingly being used in robotics and mobile platforms. Despite its lower accuracy, compared to iris and fingerprint recognition technologies, it is widely used in security systems due to the features of the contactless identification process (Engelbrecht, 2007). The general scheme of recognition is presented in Fig. 1.

Figure 1 – Process recognition scheme



Source: Prepared by the authors.

Facial recognition comprises two main stages. The first stage involves selecting facial features, while the second stage classifies the identified objects. Each human face has numerous nodal points used as distinctive features, such as the distance between the eyes, nose width, depth of eye sockets, cheekbone shape, and jawline length. These nodal points are measured to generate a numerical code called a "faceprint," which represents the face in the database (Engelbrecht, 2007).

In the past, facial recognition software relied on 2D images to compare or identify other 2D images in a database. To be effective and accurate, the image needed to capture the face nearly straight-on to the camera, with slight variation in light or facial expression compared to the database image. This posed a significant challenge, as photographs were often not taken in controlled environments. Even minor variations in lighting or orientation could degrade system performance, resulting in difficulties in finding accurate matches in the database and leading to high error rates (Engelbrecht, 2007).

Historically, law enforcement agencies primarily used facial recognition software, which utilized these systems to identify individuals in crowds. Some governmental agencies also employ these systems to enhance security and prevent electoral fraud. An example is the US-VISIT program by the U.S. government, which applies this technology to foreign visitors admitted to the country. When a foreign tourist receives a visa, their fingerprints and photographs are recorded. These data are then checked against a database of known criminals and suspected terrorists. When travelers arrive at a checkpoint in the U.S., their fingerprints and photographs are again used to verify their identity (Engelbrecht, 2007).

Scientific research explores issues related to facial recognition based on invariant moments (Reinartz, 2002; Jankowski; Grochowski, 2004). Invariant moments are often used as features in person identification tasks. The work by Jankowski and Grochowski (2004) investigates the properties of these moments, demonstrating that different datasets exhibit varying sensitivities to changes. Despite existing approaches to selecting facial features such as lips, nose, and facial profile, considering various complicating factors in image analysis (such as noise, variations in facial orientation, and emotional expressions), there is still no universally effective approach to solving these challenges.

A combined approach is proposed that integrates various tools for facial recognition. This includes methods for selecting invariant moments, forming reference classes of people, the Euclidean-Mahalanobis metric, and the use of artificial neural networks (Reinartz, 2002). Generalized transformations modified for processing three-dimensional images with unknown rotation and scale parameters have been addressed in various studies (Hart, 1968). Algorithm results for rapid object detection using the cascade method of simple functions and methods for real-time facial recognition are discussed (Gates, 1972; Aha; Kibler; Albert, 1991).

Among recent studies dedicated to facial recognition, notable works include those by Wilson (1972) and Kibler and Aha (1987), who developed theoretical probabilistic models for halftone images and applied methods for person identification based on the Bayesian rule. Tomek (1976) emphasizes the importance of researching modern facial recognition systems through in-depth studies of machine learning problems.

The study of issues related to computer image recognition tools and their visualization based on received data includes the construction of histograms of oriented gradients for person identification (Jankowski, 2000). Broadley (1993) and Wilson (2000) explore effective image segmentation based on graphical representation. The use of neural networks for facial recognition is extensively investigated in the work by Ritter *et al*. (1975).

Principal methods and techniques for gesture representation and recognition, as well as methods for dynamic facial recognition using video content studies, are defined (Skalak, 1994; Domingo; Gavalda; Watanabe, 1999). Key design stages for developing and implementing video surveillance and monitoring systems have been established (Kohonen, 1988; Madigan *et al*., 2002). Issues related to visual activity recognition and interactions using stochastic analysis, including the projection of adaptive background image models for real-time detection and tracking of people, are addressed in various scientific studies (Suykens; Vandewalle, 1999; Reeves; Bush, 2001; Li *et al.,* 2007; Sane; Ghatol, 2007; Evans, 2008; Koskimaki *et al.*, 2008; Subbotin, 2013a, 2013b).

Software is growing in popularity in various applications as systems become more accessible and their use becomes widespread. These systems are already integrated with cameras and computers used in banks, airports, and other social environments. The United States Transportation Security Administration (TSA) is currently testing software for frequent flyers who need to register on the platform. This software enables quick screening of passengers and assesses security threats. Queues have been organized into two columns to optimize airport flow, with one utilizing facial biometrics for passenger verification.

Other applications integrate TSA functionalities with money transfer services. These programs can swiftly verify a customer's identity by obtaining authorization to store a digital image. The FaceID software generates a facial imprint from this image to safeguard customers against identity theft. The use of facial recognition eliminates the need to present a photo ID, bank card, or Personal Identification Number (PIN) to verify the customer's identity, assisting businesses in fraud prevention.

While all the examples mentioned above operate with human permission, there are also systems that do not require authorization. Sometimes, the system may photograph the customer without permission, thus violating privacy laws. It is clear that these technologies have often faced criticism due to uncertainties about their impact on an individual's life. The harm caused by facial recognition technologies to privacy, freedom of expression, and due process affects us all and should not be taken lightly. Even if facial recognition technology did not have issues of biased accuracy or were not deployed randomly and carelessly, thus increasing the likelihood of errors, it would still pose a serious threat to democratic values, functioning exactly as intended.

This study investigates the processes of recognition and image processing widely used in modern security information systems for person recognition. During the execution of assigned tasks, a detailed analysis of biometric facial recognition using the FaceID system was conducted, exploring different facial recognition methods and investigating their scope of implementation and application areas. A significant contribution of the research was developing and evaluating a facial recognition system based on the Viola-Jones contour selection method, considered effective and suitable for implementation. The authors paid special attention to the functionality of the OpenCV computer vision library, emphasizing the need for further in-depth investigations.

Materials and methods

The aim of this study is to determine the feasibility of implementing a facial recognition system using computer vision and facial recognition libraries. Research tasks include studying facial recognition technology, analyzing detection methods, recognition, formation of feature points, and facial feature encoding based on an electronic portrait. The intended outcome of this work is the development of a system capable of identifying a person through their human face.

Among the most promising biometric recognition methods, facial recognition stands out. This technique offers several advantages over similar methods, including high accuracy in identification, remote verification capability, anonymous analysis, and the requirement of only a video camera. The diversity of available algorithms, combined with speed and accuracy in searching, enables the system to operate effectively under various conditions. These characteristics have driven the development of the method, making it the second most common after fingerprinting.

To further enhance accuracy, a combination of multiple facial analysis algorithms is essential. For example, ear identification effectively complements facial identification with proven efficacy. However, improper optimization when using multiple algorithms can neutralize the benefits of this combined approach. One of the most promising trends in the biometrics market is the introduction of smart digital cameras with integrated facial analysis capability. These cameras offer high image quality and can attach metadata to images containing information about detected faces. This reduces hardware overhead, lowers the cost of biometric recognition systems, and makes them more accessible. Additionally, compressed data transmission and a small stream of face-detection images alleviate data channels.

The choice of the Viola-Jones method by the researchers in this study is based on its proven efficiency in searching for objects in real-time images and videos. Studies conducted by national and international scientists demonstrate that this method has a low probability of error in identifying individuals. This method's recognition accuracy is considerably high, which is a desirable outcome. However, it is essential to note that the standard method is unable to detect human faces turned at arbitrary angles, which may limit its application in modern production systems, considering increasing technological demands (Winarno *et al.,* 2018).

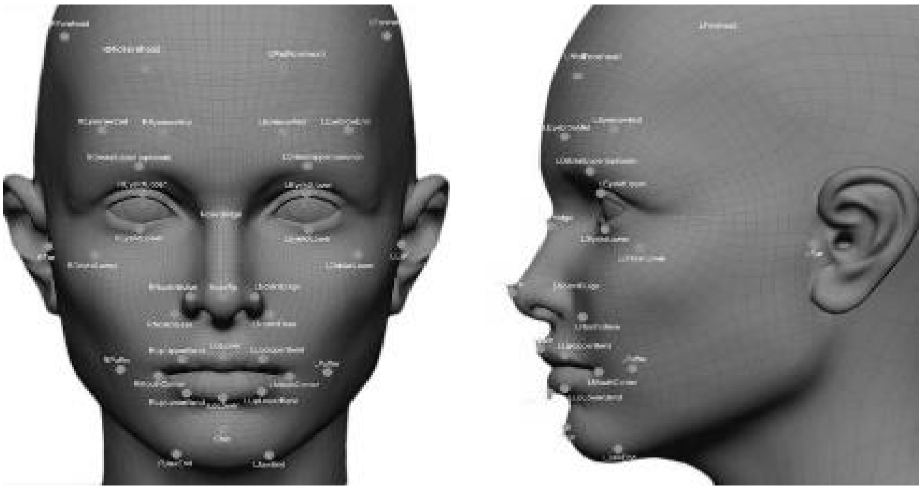
Facial recognition systems are widely used not only for purposes such as identifying criminals or wanted individuals in crowded places but also for everyday household tasks. With the proliferation of cameras and the continuous improvement of facial detection and analysis algorithms, recognition accuracy has significantly increased. Despite functional similarities among available software, users' choices are often based on individual preferences.

Results and Discussion

High-quality facial recognition significantly depends on the conditions under which the system is implemented. It is crucial to establish these conditions to ensure proper system performance. Most modern facial recognition systems can operate efficiently under specific conditions. For instance, it is essential to organize the flow of people at checkpoints to allow effective short-term facial capture. Additionally, the positioning of cameras in relation to the face should not vary by more than 30 degrees. Strict adherence to these conditions is paramount to achieving accurate identification and effective individual search, as indicated by the high precision rates reported by manufacturers of these systems.

Recently, there has been a trend towards the use of facial recognition software employing 3D models to enhance accuracy. 3D facial recognition utilizes specific facial features such as the contours of eye sockets, nose, and chin to capture a real-time three-dimensional image of a person's face (Figure 2). These features are unique to each individual and remain consistent over time (Engelbrecht, 2007).

Figure 2 – Three-dimensional face model



Source: Prepared by the authors

Using a depth axis and measurement unaffected by lighting, 3D facial recognition can be employed even in low-light environments, allowing recognition of the object from different viewing angles, including up to 90 degrees (profile view). With the use of software featuring 3D recognition capabilities, the system executes a series of steps to verify a person's identity:

- **Detection** - the image can be obtained by scanning an existing photograph (2D) digitally or capturing a live image of the object through video (3D);
- **Alignment** - after detecting the person, the system determines the position, size, and posture of the head. In the case of 3D recognition, the system can identify the object at an angle of up to 90 degrees, whereas in 2D models, the head must be turned towards the camera by at least 35 degrees;
- **Measurements** - the system performs calculations of the person's curves with submillimeter or microwave precision, thus creating a detailed model;
- **Encoding** - the system translates the model into a unique code. This encoding assigns a set of numbers to each model representing the facial features of the object;
- **Matching** - if the image is three-dimensional and the database contains three-dimensional images, the comparison is conducted without needing alterations to the

image (Hart, 1968; Wilson, 1972; Tomek, 1976; Kibbler; Aha, 1987; Jankowski, 2000; Reinartz, 2002; Jankowski; Grochowski, 2004; Engelbrecht, 2007).

There is a significant challenge with authenticating images that remain in 2D format. In contrast, within the context of 3D recognition, which involves representing an object in three spatial dimensions typically through X, Y, and Z coordinates, distinct points such as the outer eye corner, inner eye corner, and nose tip are determined. These points are marked and measured to create a 3D image. Subsequently, an algorithm is applied to project this image into a 2D format. After this conversion, the program compares the resulting image with the 2D images stored in the database to identify potential matches.

There are two main modes of operation: verification and identification. In verification (1:1), the image is exclusively compared with a specific figure in the database to validate identity. In contrast, in identification (1:2), the image is compared with multiple figures in the database, generating a score for each potential match. This method is used when it is necessary to identify a person among several.

The vector model is reduced and primarily used for fast searches in databases, especially in one-to-many searches. Surface Texture Analysis (ATS) is the largest of the three methodologies and performs a final step after the LFA model search, relying on skin elements in the image that contain more detailed information. Due to the combination of all these patterns, Face ID holds an advantage over other systems. It is insensitive to changes in facial expression, including blinking, frowning, or smiling, and can compensate for the presence of a beard, mustache, and wearing glasses. However, Face ID is not flawless. Several factors can hinder recognition, such as significant reflections on glasses, the presence of sunglasses, hair covering the central part of the face, inadequate lighting resulting in poorly illuminated images, and low resolution (images captured from far away).

Nevertheless, manufacturers are striving to improve the usability and accuracy of systems. Facial recognition systems are used in various areas:

- *Phone unlocking:* Many smartphones, including the latest iPhones, use facial recognition to unlock the device.
- *Law enforcement:* Facial recognition is regularly used by security agencies.
- *Airports and border control:* The technology has become common in many airports around the world.

- *Missing persons search:* It can be used to locate missing persons and victims of human trafficking.
- *Retail crime reduction:* It is employed to identify known thieves, organized retail criminals, or individuals with a history of fraud entering stores.
- *Enhancing shopping experience:* The technology has the potential to improve the retail customer experience.
- *Banks:* Biometric online banking is another benefit of facial recognition.
- *Marketing and advertising:* Marketing professionals use technology to enhance consumer experience.
- *Healthcare:* Hospitals use facial recognition to assist patients.
- *Attendance control for students or employees:* Some educational institutions use technology to ensure students' attendance in classes (Hart, 1968; Reinartz, 2002; Jankowski; Grochowski, 2004; Engelbrecht, 2007; Subbotin, 2013a, 2013b).

In addition to these applications, IP camera management software for PC workers (NVR) is supported by an authentic central control system, which is a monitoring and control solution that supports an unlimited number of cameras. The main console is the NVR's recording server, which displays live video and configures the system. The Save Video feature converts images to standard video formats. To ensure brightness, sharpness, and uniform grayscale in the image, a video enhancer is required. Another significant use of video analytics in surveillance systems is the use of intelligent cameras. A smart camera is equipped with an additional module that performs video processing, and these two elements are typically integrated into a single body.

The primary difference between an intelligent camera and a regular camera is that the smart camera analyzes what it sees and makes decisions based on the results of that analysis. A computer program performs mathematical analysis of input data and identifies patterns that can be described mathematically. A person's unique facial features are encoded into a computer file using only a tiny amount of memory (less than 100 bytes). This face is compared to previously captured and stored in a database. The operator working with the data should be informed by displaying additional information about the current video stream. Typical disadvantages of such systems include limited functionality, inability to expand software and hardware, and data processing in corporate "clouds" (Subbotin, 2013a, 2013b).

Currently, the company is investing in research and development to expand its business capabilities in access control and security alarm systems. The advantages of this system include: 1080p video; Instant notifications of activities; Two-way audio; Perfect image up to a distance of 500 meters, without loss of quality; Operation in darkness; and the ability to define activity zones (Subbotin, 2013a, 2013b). Based on the research, a table of functional requirements was compiled (Table 1).

Table 1 – Functional requirements of the program

1.	*The program must load the face base*
2.	The program should extract face signatures from the names of photos in the database
3.	The program should output the image from the web camera to the monitor
4.	The program should display the signature of the recognized face or the word «Unknown» if the face is not recognized

Source: Prepared by the authors.

The choice of category and method depends on the constraints and conditions of person recognition. Among the constraints influencing the choice of principle to solve the problem, the following stand out: the presence or absence of artificial obstacles on the face, spatial characteristics of people's positions, image color, face scale and resolution, number of people in the image, object lighting conditions, and the priority to minimize false recognitions or maximize the number of recognized individuals.

Various methods and approaches are used in facial recognition systems. Among them, Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Hidden Markov Models (HMM), and Gabor Wavelets are prominent (Barina, 2011; Yoon, 2009; Jurafsky; Martin, 2016). When using Hidden Markov Models to solve the facial recognition problem, each face class computes its own hidden Markov model. Then, for the unknown method, all available models are executed, and the one that provides the closest result is sought. The drawback of this approach is that Hidden Markov Models do not have good resolution, as the learning algorithm maximizes the response for its classes but does not minimize the response for other classes.

Recognition methods based on Gabor Wavelets show high efficiency. Gabor filters are used in the preprocessing stage to form a Gabor feature vector of a facial image. The Gabor Wavelet method is robust against changes in illumination because it does not directly use the grayscale values of each pixel but extracts features (Hart, 1968; Aha; Kibler; Albert, 1991; Reinartz, 2002; Jankowski; Grochowski, 2004).

Next, the paper describes and analyzes modern facial recognition methods, starting with Principal Component Analysis (PCA). The idea is to represent facial images as a set of principal components of the images, known as "Eigenfaces." These faces have the useful property that each corresponding vector resembles the shape of a face. The calculation of principal components is achieved by obtaining the eigenvectors and eigenvalues of the covariance matrix derived from the image. Image reconstruction is obtained by the linear combination of the principal components multiplied by the corresponding eigenvectors (Hart, 1968; Jankowski; Grochowski, 2004).

For each facial photograph, its principal components are calculated, typically ranging from 5 to 200. The recognition process involves comparing the principal components of the unknown portrait with the components of all known images. It is assumed that images corresponding to the same person form clusters in the feature space. The photos with the smallest distance to the input (unknown) image in the database are selected (Jankowski; Grochowski, 2004).

The method of own faces requires idealized conditions for its application, such as uniform lighting parameters, neutral facial expression, and the absence of obstacles such as glasses and beards. If these conditions are not met, the principal components will not reflect interclass variations. So, under different lighting levels, this method is practically useless, since the first principal components mainly reflect changes in lighting, and the comparison gives images with the same lighting level. Under idealized conditions, the recognition accuracy using this method can reach a value of more than 90%, which is an excellent result.

Calculation of a set of eigenvectors is highly time-consuming. One of the methods is to collapse images by rows and columns – in this form, the presented image is smaller, and the calculation and recognition process is faster, but restoring the original image is no longer possible. Viola-Jones method. This method is highly effective for searching for objects in images and video sequences in real-time. This solution has an extremely low probability of falsely identifying a person. The detector works perfectly and finds facial features even when observing the object at a small angle, approximately up to 30 degrees. The recognition accuracy using this method can reach over 90%, which is an excellent result (Winarno *et al.,* 2018).

Comparison of templates (Template Matching). The basis of this method is the selection of areas of the face in the image, and the subsequent comparison of these areas for two different images. Each similar region increases the degree of image similarity. The simplest algorithms, such as pixel comparison, are used to compare regions (Hart, 1968; Reinartz, 2002; Jankowski;

Grochowski, 2004). The disadvantage of this method is that it requires a lot of resources both for storing plots and for comparing plots. Considering the fact that with the help of the simplest comparison algorithm, the picture must be taken under strictly established conditions: it is impossible to allow noticeable changes in the course, education, mental expression, etc. The accuracy of recognition using this method is about 80%, which is a good result (Jankowski; Grochowski, 2004).

Hopfield neural network. The Hopfield network learning algorithm differs significantly from classical perceptron learning algorithms in that, instead of successively approaching the desired state with error calculations, all coefficients of the weight matrix are calculated using a formula, in just one cycle, after this procedure the network is ready for work. Limitations of the method: the images should not be very similar for memorization; the image must not be shifted or rotated relative to its original state. To eliminate these shortcomings, various modifications of the classical Hopfield neural network are considered. This network with an orthogonal transformation allows you to restore highly correlated images by processing their original set into a dual set of vectors. Thus, a neural network is created, which has the ability to remember several vectors, and when any vector is input, it can determine which of the memorized ones it is most similar to. The accuracy of recognition using this method is more than 90%, and in some cases, it even approaches 100%, which is an almost excellent result (Hart, 1968; Aha; Kibler; Albert, 1991; Broadley, 1993; Jankowski, 2000; Wilson, 2000; Reinartz, 2002; Jankowski; Grochowski, 2004).

A comprehensive approach to facial recognition, which has proven promising in addressing the proposed task, involves the implementation of computational experiments using halftone images through Euclidean-Mahalanobis classifiers (based on metrics), probabilistic neural networks, and invariant moments. The use of Euclidean-Mahalanobis metrics allows the system to handle variations in head orientation and changes in image brightness. On the other hand, probabilistic neural networks effectively manage challenges such as closed eyes and variations in facial expressions (such as smiles and frowns). However, training artificial intelligence for neural networks requires significant time investment due to the extensive amount of data required.

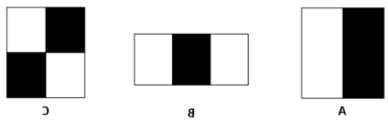
A significant advantage of facial recognition technology should be highlighted: its inviolability, as it does not involve elements like passwords that can be stolen or altered. It is not feasible to ensure that a person maintains a fixed position to remain still and look directly at the camera, which occasionally compromises result accuracy. If a person changes their appearance, such as altering their hair or wearing accessories, recognizing them can become practically impossible (Broadley, 1993; Jankowski, 2000; Wilson, 2000; Reinartz, 2002; Jankowski; Grochowski, 2004).

Based on the above considerations, the development of hybrid methods that combine the advantages and minimize the disadvantages of the mentioned individual approaches seems to be a promising direction for the continued advancement of facial recognition technology.

A particularly notable algorithm is Viola-Jones. Although slow for learning, this algorithm can detect faces in real-time with impressive efficiency. Operating on grayscale images, the algorithm examines various smaller sub-regions of the image and searches for specific features in each to identify faces. It requires checking many different positions and scales, as an image may contain multiple faces of varying sizes. Viola and Jones used Haar-like features for face detection in this algorithm (Reinartz, 2002; Jankowski; Grochowski, 2004).

It is important to note that this method presents a very low probability of false identification of a person. However, the detection effectiveness significantly decreases at angles greater than 30 degrees, making it challenging to detect human faces positioned at variable angles. This aspect can complicate the implementation of the algorithm in modern production systems, considering the increasing demands for accuracy. The algorithm relies on three types of Haar-like features, as illustrated in Figure 3.

Figure 3 – Three types of Viola-Jones characteristics

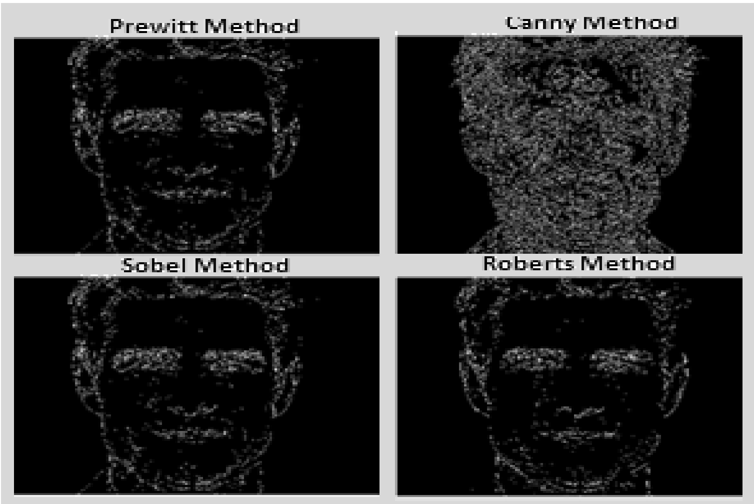


Source: Prepared by the authors.

The integral representation of the image is a matrix that coincides in size with the original image. Each of its elements stores the sum of the intensities of all pixels located to the left and above the given element. The wavelet transform is often used to analyze unstable processes. Such a tool has shown its effectiveness in a wide class of tasks related to image processing. Wavelet transform coefficients hold information about the analyzed process and the used wavelet. Therefore, the choice of wavelet for analysis is determined by what information needs to be extracted from the process. Each wavelet has characteristic features during rectilinear movement in the temporal and frequency domains.

To determine the contours of the face, several experiments were conducted in MatLab – a system of computer mathematics, for which the CV (Computer vision) and Toolbox libraries were connected. Results after the experiments: the method of Prewitt and Sobel turned out to be the most effective within the given limits of application (Fig. 4); the percentage of finding a face is 75%; the slope percentage is from 14% to 26%. These experiments showed that the most effective methods of boundary selection are the methods of Sobel and Prewitt. They differ from other methods in that they will reflect the main and additional contours of the image without oversaturating the image with noise. These methods will be used in the future.

Figure 4 – Contour selection methods



Source: Prepared by the authors.

For the «Face recognition» system, the Python programming language was chosen, and in this language, such a library as «OpenCV» is an open-source computer vision library designed for image analysis, classification, and processing. Widely used in languages such as C, C++, Python, and Java. The library is used to detect objects, faces, diseases, lesions, license plates, and even handwritten text in various images and videos. With OpenCV in Deep Learning, we expand the vector space and perform mathematical operations on these features to identify visual patterns and their various characteristics. Computer vision can be defined as a discipline that explains how to reconstruct, extract, and understand a 3D space from its 2D images in terms of the properties of the structure present in the space. It is engaged in modeling and replicating human vision using computer software and hardware (Wilson, 1972).

Computer vision is the process by which we can understand images and videos, how they are stored, and how we can manipulate and extract data from them. Computer vision is the basis of, or primarily used for, artificial intelligence. «Computer-Vision» plays a vital role in self-driving cars, robotics, and photo-correction applications. Computer vision, unlike photography, does not reflect reality, but instead interprets and misinterprets it by imposing meaning on statistical assumptions. There is no true value in the raw data of computer vision algorithms, only statistical probabilities truncated into Boolean states masquerading as true results with added meaning in post-production. Let's analyze other methods for face recognition. Dlib is one of the most powerful and easy-to-use Open-source libraries consisting of machine learning libraries/algorithms and various software development tools. Open-source licensing of Dlib allows you to use it in any application for free (Tomek, 1976).

Dlib offers two different functions for face capture. In particular, HoG + Linear SVM – the Histogram Oriented Gradients (HoG) + Linear Support Vector Machine (SVM) algorithm in Dlib offers very fast frontal face recognition, but has limited capabilities for recognizing facial poses at sharp angles (such as CCTV footage or random observation environment, where the subject does not take an active part in the identification process). It also supports passport profile faces, albeit with a very small margin of error (faces pointing up or down, etc.). HoG + SVM is suitable for limited situations where the sensor can rely on a direct and unobstructed view of the participant's face, such as ATMs and mobile frame identification systems, as well as mobile CCTV recognition systems where cameras can obtain a direct snapshot profile of drivers.

CNN Max-Margin Face Detector (MMOD) MMOD is a robust and reliable GPU-accelerated face detector that uses a Convolutional Neural Network (CNN) and is much better at detecting faces at obscure angles and in challenging environments, suitable for casual surveillance and urban analysis.

MMOD is not a stand-alone alternative to HoG + Linear SVM, but rather can be applied to HoG itself or any visual bag-of-words model that treats detected clusters of pixels as probe entities for potential labeling – including face recognition. Let's compare HoG & MODD. The appeal of HoG + Linear SVM under Dlib is its low resource usage; its effectiveness when working on the CPU; the fact that it has at least some width for non-frontal faces; its requirements for a low-impact model; and a relatively robust occlusion detection procedure. On the other hand, the default deployment requires a minimum face size of 80x80 pixels. If you need to detect faces below this threshold, you will need to train your implementation. In addition, this approach produces poor results for sharp corners of the face; creates bounding frames that can excessively crop facial features; and combats complex cases of occlusion.

The advantage of MMOD (CNN) under Dlib is (perhaps primarily) its ability to recognize complex face orientation (which may be a deciding factor, depending on your target environment); its impressive speed with access to even an average GPU; easy learning architecture; and its excellent occlusion processing. On the other hand, it can produce bounding rectangles even more constrained than HoG + Linear SVM in the standard deployment; runs significantly slower on CPU than HoG/LSVM; and shares HoG/LSVM's inability to detect faces smaller than 80 square pixels – again, requiring a custom build for certain scenarios, such as sharp street vantage points that extend into the distance.

The main task of the «Face Recognition» system is face detection. Because you need to find a face in a photo or picture before you can distinguish them, you can also use the smartphone camera, and in this mode, the program will be able to recognize faces. Face recognition is a function of cameras. When the camera can automatically detect faces, it understands that all faces are in focus before taking the picture. In this case, such a function will be used for another purpose – to detect a part of the image that will later be used for recognition. This work chooses the more reliable method, «Histogram of Oriented Gradients» – or simply HOG –. To find a face in an image, you need to make it black and white, since color data is not needed for this process. A function in the OpenCV library is used to change the color segment of a picture or photo. Thus, this function converts a photo or picture into black-and-white format (Fig. 5).

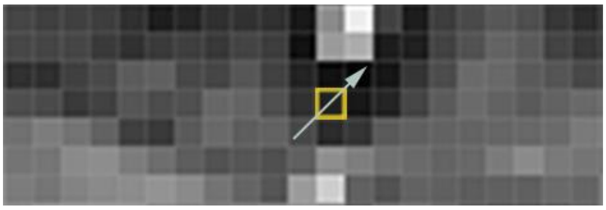
Figure 5 – Converting the image to black and white

```
def findEncodings(images):
    encodeList = []
    for img in images:
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
```

Source: Prepared by the authors.

Each pixel in the image is then examined one at a time. In this case, it is necessary to consider all the pixels that are around the lonely pixels. The main task of this process is to find information about the darkest pixel in the image and collect data about other pixels around it. The next step will be an arrow indicating the direction in which the picture becomes darker (Fig. 6).

Figure 6 – Pixel gradients



Source: Prepared by the authors.

Repeating this procedure for each pixel in the image will eventually replace all pixels with an arrow. These arrows are called gradients, and they show the flow from light to dark pixels throughout the image. It is worth replacing pixels with gradients. Based on the image comparison analysis, it can be seen that both dark and very light photos of the same person will have different pixel values. But if you take into account the direction of the change in brightness in the photo, which light and dark images will ultimately result in the same image. The disadvantage of this solution is that, as a result, we will get a large number of gradients, which will complicate further face recognition.

Because of this, the image is divided into squares with an area of 256 pixels, that is, a square with sides of 16 by 16, the next step calculates the number of gradients indicating the direction of brightness. It is calculated which number points to the top, to the upper right corner, to the right, and so on. Then, in this square, the gradient is replaced by the directions of the arrows, which were in the majority. As a result, we get a simple image in which the main structure of the face is visible. To find a face in an HOG image, all one needs to do is find the part of another image that looks most similar to the now-known HOG pattern drawn from a bunch of other training faces (Figure 7).

Figure 7 – Comparison of features

Source: Prepared by the authors.

To fulfill the previous conditions, you need to use the function for finding faces in the «face_recognition» library, which uses the detection method (Fig. 8). The technology makes it easy to find faces in any image.

Figure 8 – Face detection

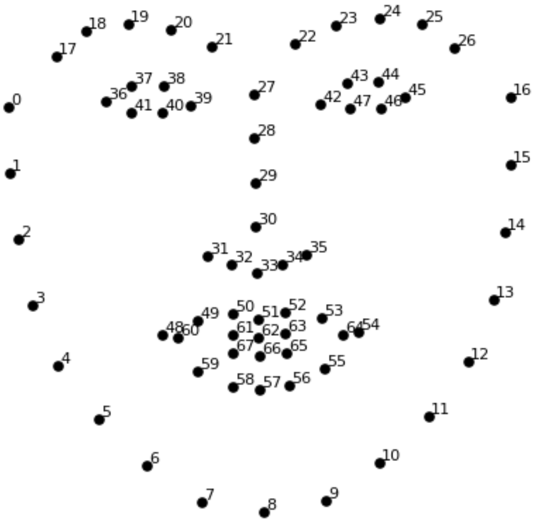
```
faceLoc = face_recognition.face_locations(imgRus)[0]
encodeElon = face_recognition.face_encodings(imgRus)[0]
cv2.rectangle(imgRus, (faceLoc[3], faceLoc[0]),
              (faceLoc[1], faceLoc[2]), (255, 0, 255), 2)

faceLocTest = face_recognition.face_locations(imgTest)[0]
encodeTest = face_recognition.face_encodings(imgTest)[0]
cv2.rectangle(imgTest,(faceLocTest[3],faceLocTest[0]),
              (faceLocTest[1],faceLocTest[2]),(255,0,255),2)
```

Source: Prepared by the authors.

The next task is the formation of a facial landmark. The face in the image is isolated. From that moment, a problem appeared with the human face, which can be turned in different directions, and because of this, it looks different to the AI. In order to solve this problem, you need to make sure that the lips and eyes are aligned in the image in the same way as in the sample. This will significantly simplify the comparison of faces in the next steps. For this, an algorithm called «face landmark estimation» is used. Of course, there are many methods for solving the given problem, but Kazema and Sullivan's method will be used in this situation. The basic idea is to highlight 68 specific points (called landmarks) that exist on every face – the top of the chin, the outer edge of each eye, the inner edge of each eyebrow, and so on. After that, the algorithm needs to be programmed so that it can find 68 specific points on different face shapes and types (Fig. 9).

At this point, after we've trained the algorithm to recognize faces and mouths, all that's left to do is rotate, scale, and pan the image so that the eyes and mouth are as centered as possible. In an affine transformation, all parallel lines in the original image will remain parallel in the original image.

Figure 9 – Key points of the face «Landmarks»



Source: Prepared by the authors.

To find the transformation matrix, we need three points from the input image and their corresponding locations in the output image. An affine transformation uses a clockwise rotation angle, as opposed to the typical geometry unit circle, which is measured by rotating counter-clockwise from 0, starting on the positive X-axis, so you'll often see a negative angle value used.

An affine transformation is one that has the following properties: any affine transformation can be represented as a sequence of operations from among the simplest ones: shift, stretching/compression, rotation; straight lines, the parallelism of consecutive lines, the ratio of the lengths of segments lying on the same line, and the ratio of the areas of figures are preserved. New coordinates f (x), which have the same position in space relative to the «new» coordinate system, which coordinates x had in the «old». Therefore, it will be enough to use the fundamental transformations: scale and rotation, but they must preserve parallel lines, that is, affine loops. So, we have completed one of the main tasks, namely the centering of the eyes and mouth, regardless of the angle of the face in the image, the algorithm will center plus-minus in the same position. This is important for the accuracy of the following steps.

In order to identify a person, we use the approach of comparing an unknown face according to the features we have already described, with faces already in the collected photo database. Now, we need the unknown face to not just look like the person already in our database; it has to be that person. This approach is very problematic. A website like Facebook, with billions of users and trillions of photos, can't go through every previously tagged face to compare it to every newly uploaded image. It would take too much time. In contrast, algorithms must be able to recognize faces in milliseconds, not hours. In order not to overload the device for comparing faces, we need to choose the main parameters of some parts of the face that will be compared by calculation. In order to determine the most accurate approach to face comparison, the researcher conducted a series of experiments and found that in this situation, it is better to allow the computer to independently choose which measurements to collect. In this, the algorithm performs the task better than a human.

The solution is to train a deep convolutional neural network. Now, we need to train the network to generate 128 values for any face, which is better and more efficient than training it to recognize faces. The network is trained when three types of face images are viewed simultaneously: a training image of a famous person's face, another image of the same famous person, and an image of a completely different person. After that, the algorithm looks at the values it generates in the process for all three images.

As a next step, the algorithm adjusts the neural network to ensure that the measurements generated for the first and second images are more similar to each other than the measurements between images. By repeating this step millions of times for images of thousands of different personas, this neural network learns to generate 128 parameters for each persona. Ten different photos of the same person should give similar measurements. People who study machine learning call the 128 parameters of each face «embedding». In machine learning, the idea of reducing raw data, such as images, is prevalent. Encoding our face image is a process of training a convolutional neural network to extract embedded faces, which requires a lot of data and computer power. Even with an expensive NVidia video card, Telsa requires about 24 hours of continuous training to get good accuracy (Jankowski, 2000).

After training, the network has the ability to generate measurements for all types of faces, even if it has never seen one. So this step only needs to be done once. Thanks to OpenFace employees who were trained and trained, their work was published by implementing several trained networks, and one of those networks was used in this work. You then pass your face images through the pre-trained network to get 128 measurements for each face (Figure 10).

Figure 10 – Face coding

0.097496084868908	0.045223236083984	-0.1281466782093	0.032084941864014
0.12529824674129	0.060309179127216	0.17521631717682	0.020976085215807
0.030809439718723	-0.01981477253139	0.10801389068365	-0.00052163278451189
0.036050599068403	0.065554238855839	0.0731306001544	-0.1318951100111
-0.097486883401871	0.1226262897253	-0.029626874253154	-0.0059557510539889
-0.0066401711665094	0.036750309169292	-0.15958009660244	0.043374512344599
-0.14131525158882	0.14114324748516	-0.031351584941149	-0.053343612700701
-0.048540540039539	-0.061901587992907	-0.15042643249035	0.078198105096817
-0.12567175924778	-0.10568545013666	-0.12728653848171	-0.076289616525173
-0.061418771743774	-0.074287034571171	-0.065365232527256	0.12369467318058
0.046741496771574	0.0061761881224811	0.14746543765068	0.056418422609568
-0.12113650143147	-0.21055991947651	0.0041091227903962	0.089727647602558
0.061606746166945	0.11345765739679	0.021352224051952	-0.0085843298584223
0.061989940702915	0.19372203946114	-0.086726233363152	-0.022388197481632
0.10904195904732	0.084853030741215	0.09463594853878	0.020696049556136
-0.019414527341723	0.0064811296761036	0.21180312335491	-0.050584398210049
0.15245945751667	-0.16582328081131	-0.035577941685915	-0.072376452386379
-0.12216668576002	-0.0072777755558491	-0.036901291459799	-0.034365277737379
0.083934605121613	-0.059730969369411	-0.070026844739914	-0.045013956725597
0.087945111095905	0.11478432267904	-0.089621491730213	-0.013955107890069
-0.021407851949334	0.14841195940971	0.078333757817745	-0.17898085713387
-0.018298890441656	0.049525424838066	0.13227833807468	-0.072600327432156
-0.011014151386917	-0.051016297191381	-0.14132921397686	0.0050511928275228
0.0093679334968328	-0.062812767922878	-0.13407498598099	-0.014829395338893
0.058139257133007	0.0048638740554452	-0.039491076022387	-0.043765489012003
-0.024210374802351	-0.11443792283535	0.071997955441475	-0.012062266469002
-0.057223934680223	0.014683869667351	0.05228154733777	0.012774495407939
0.023535015061498	-0.081752359867096	-0.031709920614958	0.069833360612392
-0.0098039731383324	0.037022035568953	0.11009479314089	0.11638788878918
0.020220354199409	0.12788131833076	0.18632389605045	-0.015336792916059
0.0040337680839002	-0.094398014247417	-0.11768248677254	0.10281457751989
0.051597066223621	-0.10034311562777	-0.040977258235216	-0.082041338086128

Source: Prepared by the authors.

This network generates the same numbers by comparing two different images of the same person. In order to use a neural network pre-trained on powerful equipment, the face coding function in the Face recognition library is used (Fig. 11).

Figure 11 – Face coding

```
faceLoc = face_recognition.face_locations(imgRus)[0]
encodeElon = face_recognition.face_encodings(imgRus)[0]
cv2.rectangle(imgRus, (faceLoc[3], faceLoc[0]),
              (faceLoc[1], faceLoc[2]), (255, 0, 255), 2)

faceLocTest = face_recognition.face_locations(imgTest)[0]
encodeTest = face_recognition.face_encodings(imgTest)[0]
cv2.rectangle(imgTest,(faceLocTest[3],faceLocTest[0]),
              (faceLocTest[1],faceLocTest[2]),(255,0,255),2)
```

Source: Prepared by the authors.

The last step involves the algorithm finding a person in the database of famous people who has the closest measurements to the test image. It is necessary to prepare a classifier that will analyze the text image and report which person corresponds the most. This classifier takes milliseconds to run. The result of the classifier is the person's name. Next, all the above-mentioned functions are used (Fig. 12).

Figure 12 – Face recognition

```
while True:
    success, img = cap.read()
    imgS = cv2.resize(img,(0,0),None,0.25,0.25)
    imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)
    facesCurFrame = face_recognition.face_locations(imgS)
    encodesCurFrame = face_recognition.face_encodings(imgS,facesCurFrame)
    for encodeFace,faceLoc in zip(encodesCurFrame,facesCurFrame):
        matches = face_recognition.compare_faces(encodeListKnown,encodeFace)
        faceDis = face_recognition.face_distance(encodeListKnown,encodeFace)
        matchIndex = np.argmin(faceDis)
        if faceDis[matchIndex] < 0.50:
            name = classNames[matchIndex].upper()
            markAttendance(name)
        else:
            name = 'Unknown'
        y1, x2, y2, x1 = faceLoc
        y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
        cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)
        cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)
        cv2.putText(img, name, (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)
    cv2.imshow('Webcam',img)
    cv2.waitKey(1)
```

Source: Prepared by the authors.

We can test the program at this stage and determine how accurately the AI recognizes the face. For this experiment, several images of famous people were taken. To verify the functionality of the created software product, system testing was conducted after the completion of module and integration testing, after which the software was tested together with the expected environment. Functional testing methods and some structural testing methods were used. System testing ensures that each function of the system is working properly and also checks non-functional requirements such as performance, safety, reliability, stress, and load, which in the future will provide an opportunity to improve the quality of the final product. An analysis of the defects found at the system testing stage was carried out. Before eliminating the defect, an analysis of its impact was carried out. In case the system allows, defects are documented and mentioned as known limitations instead of being addressed in case of a situation where a fix is time-consuming or technically impossible in the current design, etc. (Jankowski, 2000).

Test plans have been drawn up for testing. The list of conditions under which testing will take place: determination of the person entered in the database; identification of a person not entered in the database; identification of a person in different lighting conditions; identification of the face at various angles of the face; identification of a person with different facial expressions; identification of a person in the presence of a beard, mustache, glasses, mask. Test cases are compiled for verification, according to which the program is verified (Table 2).

Table 2 – Test to verify the program

№	Description	Action	Expected result
1.	Determination of the person entered in the database.	Wait for the recognition results.	Person recognized (name displayed).
2.	Identification of a person not entered in the database.	Wait for the recognition results.	The person is not recognized («unknown» is displayed).
3.	Identification of a person in light (daylight lamp 900Lm).	Wait for the recognition results.	The person is recognized.
4.	Identifying a person in the dark.	Wait for the recognition results.	The person is not recognized («unknown» is displayed).
5.	Face detection at a face placement angle of 16–26%.	Wait for the recognition results.	The person is recognized.
6.	Identification of the face angle of placement of the face 26–40%.	Wait for the recognition results.	The person is partially recognized.
7.	Identifying a person with a beard.	Wait for the recognition results.	The person is recognized.
8.	Identification of a person in the presence of a mustache.	Wait for the recognition results.	The person is recognized.
9.	Identification of a person in the presence of glasses with transparent lenses.	Wait for the recognition results.	The person is recognized.
10.	Identification of a person in the presence of glasses with dark lenses.	Wait for the recognition results.	The person is not recognized.
11.	Identification of a person in the presence of a medical mask covering the nose.	Wait for the recognition results.	The person is not recognized.
12.	Identification of the person in the presence of a medical mask that does not cover the nose.	Wait for the recognition results.	The person is recognized.
13.	Identification of a person in the presence of a smile.	Wait for the recognition results.	The person is recognized.

Source: Prepared by the authors.

Final considerations

The scientific novelty of the obtained results lies in the fact that the face recognition system was improved, in particular, biometric recognition by means of «FaceID» it was investigated in which spheres of activity the «Face recognition» system is used for which purpose. Each part of the system was designed and developed step by step, and the face recognition system itself was built. Of course, a check was carried out for the correct operation of the system, and for this purpose, several photos with the images of different people were selected.

This work considered the analysis of the subject area of face recognition, the relevance of this system in our time, the biometric recognition of the «FaceID» system, several different methods for recognizing faces, and investigated in which spheres of activity the «Face recognition» system is used for which purpose.

As part of the study, a number of face recognition algorithms were analyzed and based on the analysis results, it was proved that the entire face recognition system can be modeled using the Violi-Jones contour extraction method and tested with a successful recognition result of approximately 75%. The functional capabilities of the OpenCV computer vision library and other libraries are considered.

The practical significance of the obtained results is that software has been developed that implements the proposed indicators, and the conducted experiments confirm the effectiveness of the proposed development. The results of the experiment make it possible to recommend the proposed software product for use in practice, as well as to determine effective conditions of use of the proposed software product.

Each part of the system was developed step by step, and the face recognition system itself was built; of course, a check was carried out for the correct functioning of the system, and for this, several photos with the images of different persons were selected. The compiled test plan reflects the main stages of the implementation of the theoretical and practical research that was conducted. The proposed software solution can be used to implement a more powerful system, for example, a video surveillance system with intelligent recognition. Prospects for further research are to improve the proposed development to increase the efficiency of its use.

REFERENCES

AHA, D. W.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. **Machine Learning**, [*S. l.*], v. 6, p. 37–66, 1991. DOI: 10.1023/A:1022689900470.

BARINA, D. Gabor wavelets in image processing. *In*: CONFERENCE STUDENT EEICT, 17., 2011. **Proceedings** [...]. Brno: Brno University of Technology. 2011. v. 3, p. 522–526.

BROADLEY, C. E. Addressing the selective superiority problem: automatic algorithm/model class selection. *In*: MACHINE LEARNING: TENTH INTERNATIONAL CONFERENCE, Amherst, 1993. **Proceedings** […]. Burlington: Morgan Kaufmann, 1993. p. 17–24. DOI: 0.1016/b978-1-55860-307-3.50009-5.

DOMINGO, C.; GAVALDA, R.; WATANABE, O. Adaptive sampling methods for scaling up knowledge discovery algorithms. *In*: DISCOVERY SCIENCE: SECOND INTERNATIONAL CONFERENCE, 1999, Tokyo. **Proceedings** […]. Berlin: Springer, 1999. p. 172–183. DOI: 10.1007/3-540-46846-3_16.

ENGELBRECHT, A. **Computational intelligence**: an introduction. Sidney: John Wiley & Sons, 2007. 597 p. DOI: 10.1002/9780470512517.

EVANS, R. **Clustering for classification**: using standard clustering methods to summarise datasets with minimal loss of classification accuracy. Saarbrücken: VDM Verlag, 2008. 108 p.

GATES, G. The reduced nearest neighbor rule. **IEEE Transactions on Information Theory**, [*S. l.*], v. 18, n. 3, p. 431–433, 1972. DOI: 10.1109/TIT.1972.1054809.

HART, P. E. The condensed nearest neighbor rule. **IEEE Transactions on Information Theory**, [*S. l.*], v. 14, p. 515–516, 1968. DOI: 10.1109/TIT.1968.1054155.

JANKOWSKI, N. Data regularization. *In*: NEURAL NETWORKS AND SOFT COMPUTING, 5., 2000, Zakopane. **Proceedings** […]. Częstochowa: Polish Neural Networks Society, 2000. p. 209–214.

JANKOWSKI, N.; GROCHOWSKI, M. Comparison of instance selection algorithms I. Algorithms survey. *In:* ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING, 7., 2004, Zakopane. **Proceedings** […]. Berlin: Springer, 2004. p. 598–603. (Lecture Notes in Computer Science, v. 3070). DOI: 10.1007/978-3-540-24844-6_90.

JURAFSKY, D.; MARTIN, J. **Hidden Markov models**. Speech and Language Processing. Draft of November 7, 2016.

KIBBLER, D.; AHA, D. W. Learning representative exemplars of concepts: an initial case of study. *In*: MACHINE LEARNING INTERNATIONAL WORKSHOP, 4., 1987, Irvine. **Proceedings** […]. Burlington: Morgan Kaufmann, 1987. p. 24–30. DOI: 10.1016/b978-0-934613-41-5.50006-4.

KOHONEN, T. Learning vector quantization. **Neural Networks**, [*S. l.*], v. 1, p. 303, 1988. DOI: 10.1016/0893-6080(88)90334-6.

KOSKIMAKI, H.; JUUTILAINEN, I.; LAURINEN, P.; RÖNING, J. Two-level clustering approach to training data instance selection: a case study for the steel industry. *In*: NEURAL NETWORKS: INTERNATIONAL JOINT CONFERENCE, 2008, Hong Kong. **Proceedings** […]. Los Alamitos: IEEE, 2008. p. 3044–3049. DOI: 10.1109/ijcnn.2008.4634228.

LI, B.; CHI, M.; FAN, J.; XUE, X. Support cluster machine. *In:* MACHINE LEARNING INTERNATIONAL CONFERENCE, 24., 2007, Corvallis. **Proceedings** […]. New York, 2007. p. 505–512. DOI: 10.1145/1273496.1273560.

MADIGAN, D.; RAGHAVAN, N.; DUMOUCHEL, W.; NASON, M. Likelihood-based data squashing: a modeling approach to instance construction. **Data Mining and Knowledge Discovery**, [*S. l.*], v. 6, n. 2. p. 173–190. DOI: 10.1023/A:1014095614948.

REEVES, C. R.; BUSH, D. R. Using genetic algorithms for training data selection in RBF networks. *In*: **Instance Selection and Construction for Data Mining.** Norwell: Kluwer, 2001. Part VI. p. 339–356. DOI: 10.1007/978-1-4757-3359-4_19.

REINARTZ, T. A. Unifying view on instance selection. **Data Mining and Knowledge Discovery**, [*S. l.*], v. 6, p. 191–210, 2002. DOI: 10.1023/A:1014047731786.

RITTER, G.; WOODRUFF, H.; LOWRY, S.; ISENHOUR, T. An algorithm for a selective nearest neighbor decision rule. **IEEE Transactions on Information Theory**, [*S. l.*], v. 21, n. 6, p. 665–669, 1975. DOI: 10.1109/TIT.1975.1055464.

SANE, S. S.; GHATOL, A. A. A Novel supervised instance selection algorithm. **International Journal of Business Intelligence and Data Mining,** [*S. l.*], v. 2, n 4. p. 471–495, 2007. DOI: 10.1504/IJBIDM.2007.016384.

SKALAK, D. B. Prototype and feature selection by sampling and random mutation hill climbing algorithms. *In*: MACHINE LEARNING: INTERNATIONAL CONFERENCE, 11., 1994, New Brunswick. **Proceedings** […]. Burlington: Morgan Kaufmann, 1994. p. 293–301. DOI: 10.1016/b978-1-55860-335-6.50043-x.

SUBBOTIN, S. The neuro-fuzzy network synthesis and simplification on precedents in problems of diagnosis and pattern recognition. **Optical Memory and Neural Networks (Information Optics)**, [*S. l.*], v. 22, n. 2, p. 97–103, 2013a.
DOI: 10.3103/s1060992x13020082.

SUBBOTIN, S. Methods of sampling based on exhaustive and evolutionary search. **Automatic Control and Computer Sciences**, [*S. l.*], v. 47, n. 3, p. 113–121, 2013b.
DOI: 10.3103/s0146411613030073.

SUYKENS, J. A.; VANDEWALLE, J. Least squares support vector machine classifiers. **Neural Processing Letters**, [*S. l.*], v. 9, n 3. p. 293–300, 1999.
DOI: 10.1023/A:1018628609742.

TOMEK, I. An experiment with the edited nearest-neighbor rule. **IEEE Transactions on Systems, Man and Cybernetics**, [*S. l.*],v. 6, p. 448–452, 1976.
DOI: 10.1109/TSMC.1976.4309523.

WILSON, D. L. Asymptotic properties of nearest neighbor rules using edited data. **IEEE Transactions on Systems, Man, Cybernetics**, [*S. l.*], v. 2, n. 3, p. 408–421, 1972.
DOI: 10.1109/TSMC.1972.4309137.

WILSON, D. R.; Martinez, D. R. Reduction techniques for instancebased learning algorithms. **Machine Learning**, [*S. l.*],v. 38, n. 3, p. 257–286, 2000. DOI: 10.1023/A:1007626913721.
WINARNO, E.; HADIKURNIAWATI, W.; NIRWANTO, A.; ABDULLAH, D. Multi-view faces detection using viola-jones method. **Journal of Physics Conference Series**, [*S. l.*], v. 1114, n. 1, 2018. DOI: 10.1088/1742-6596/1114/1/012068.

YOON, B. J. Hidden Markov models and their application in the analysis of biological sequences. **Current Genomics**, [*S. l.*], v. 10, n. 6, p. 402–415, 2009.
DOI: 10.2174/138920209789177575.

CRediT Author Statement

Acknowledgements: We thank the National University «Yuri Kondratyuk Poltava Polytechnic».

Funding: Not applicable.

Conflicts of interest: There are no conflicts of interest.

Ethical approval: Not applicable.

Data and material availability: Not applicable.

Authors' contributions: **Alla KAPITON**: Data analysis and interpretation. **Nataliia KONONETS**: Conception, ideation, writing and revision. Volodymyr MOKLIAK: Data analysis and interpretation. **Valentyna ONIPKO**: Data collection. **Serhiy DUDKO**: Data collection. **Vadym PYLYPENKO**: Collaboration in article writing and proofreading. **Anna SOKIL**: Collaboration in article writing and proofreading.

Processing and editing: Editora Ibero-Americana de Educação.
Proofreading, formatting, normalization and translation.