CEUR-WS.org/Vol-3421/short11.pdf

Methods of Face Recognition in Video Sequences and Performance Studies

Mariia Nazarkevych[1], Vitaly Lutsyshyn[1], Hanna Nazarkevych[2], Liubomyr Parkhuts[2], and Maryna Kostiak[2]

[1]Lviv Ivan Franko National University, 1 Universytetska str., Lviv, 79000, Ukraine
[2]Lviv Polytechnic National University, 12 Stepan Bandera str., Lviv, 79013, Ukraine

Abstract

A method of capturing a person's face in a video stream has been developed. The developed methods of capturing the video stream are considered. Tracking methods are used in video surveillance. Methods of video stream capture, image frame extraction, and face recognition are considered. The method of flexible comparison on graphs, the principal component method, The Viola-Jones method, Local binary patterns, and Hidden Markov models, which are used for face recognition, are considered. The library in Python DeepFace was studied. Face recognition experiments were conducted. Faces photographed in the genre of selfie, portrait, and documentary photography were recognized. It has been found that the best recognition methods are found in the genre of photography. The recognition results are somewhat worse for selfies. The worst ones are for digital photography. Recognition was based on the MediaPipe Face Detection library. The recognition time was from 10 to 22 mc.

Keywords

Face recognition, object tracking, machine learning

1. Introduction

Tracking objects in surveillance camera footage is a challenging task. It is much more difficult to track objects in video sequences to improve their recognition. There are many existing object-tracking methods, but all have some drawbacks. Some of the existing object-tracking models are region-based contour models [1]. Tracking—tracking an object in a video sequence; and detection—detecting an object in a video sequence. Tracking-by-detection—trackers first run a detector for each frame, and then the tracking algorithm associates these detections to determine the movement of individual objects and assign them unique identification numbers [2].

Tracking objects is a complex problem. Difficulties with object tracking can arise from abrupt object movement, changing appearance patterns of both the object and the scene, non-rigid object structures, object-object and object-scene occlusions, and camera movement. Tracking is usually performed in the context of higher-level applications that require the location and/or shape of an object in each frame. Typically, assumptions are made to limit the tracking problem in the context of a particular application. In this review, we classify tracking methods based on the object and motion representations used. Object tracking consists of using appropriate image features, selecting motion models, and detecting objects [3]:

- Target representation object.
- Localization object.

Difficulties arise when objects move fast compared to the frame rate or when the tracked object changes direction in time [4–6]. The sequential flow of object detection, object tracking, object identification, and object behavior completes the tracking process [7]. Video processing consists of the following steps: video upload [8], prepro-cessing, a proposed

CPITS 2023: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2023, Kyiv, Ukraine

EMAIL: mariia.a.nazarkevych@lpnu.ua (M. Nazarkevych); vitalylutsyshyn@gmail.com (V. Lutsyshyn); hanna.ya.nazarkevych@lpnu.ua (H. Nazarkevych); liubomyr.t.parkhuts@lpnu.ua (L. Parkhuts); Kostiak.maryna@lpnu.ua (M. Kostiak)
ORCID: 0000-0002-6528-9867 (M. Nazarkevych); 0009-0008-1229-6706 (V. Lutsyshyn); 0000-0002-1413-630X (H. Nazarkevych); 0000-0003-4759-9383 (L. Parkhuts); 0000-0002-6667-7693 (M. Kostiak)

© 2023 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

algorithm that includes video processing, then the object capture step (Fig. 1).

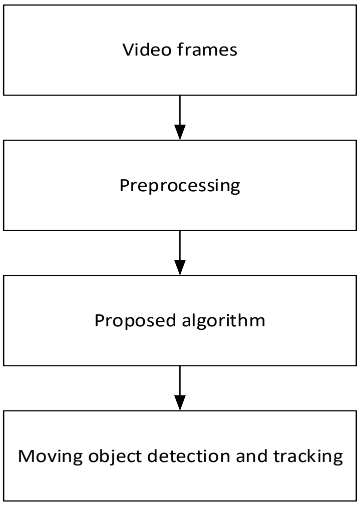

Figure 1: Scheme of video processing and object outline selection

2. Object Recognition

The capture and encoding of digital images should result in the creation and rapid dissemination of a huge amount of visual information. Hence, efficient tools for searching and retrieving visual information are essential. Although there are effective search engines for text documents today, there are no satisfactory systems for retrieving visual information.

Due to the growth of visual data both online and offline [9] and the phenomenal success of web search, expectations for image and video search technologies are increasing.

However, with the evolution of video camera characteristics that can record at high frame rates in good quality, and with advances in detection, such as new approaches based on Convolutional Neural Networks (CNNs), the basis for Tracking-by-detection trackers [10] has become more robust. The requirements for a tracker in a tracking system have changed dramatically, allowing for much simpler tracking algorithms that can compete with more complex systems requiring significant computational costs.

Let’s analyze three ranking algorithms that take into account the spatial, temporal, and spatiotemporal properties of geo-referenced video clips.

Object detection requires training machine learning models, such as Recurrent Neural Networks (RNNs) and CNNs, on images where objects have been manually annotated and associated with a high-level concept.

A video stream is streamed in which a face needs to be recognized [11]. We determine the size of the face coordinates. The face contour is aligned and the basic parameters are determined (Fig. 2). As a result, a parametric vector is built. The parameters are compared. As a result, recognition is performed.

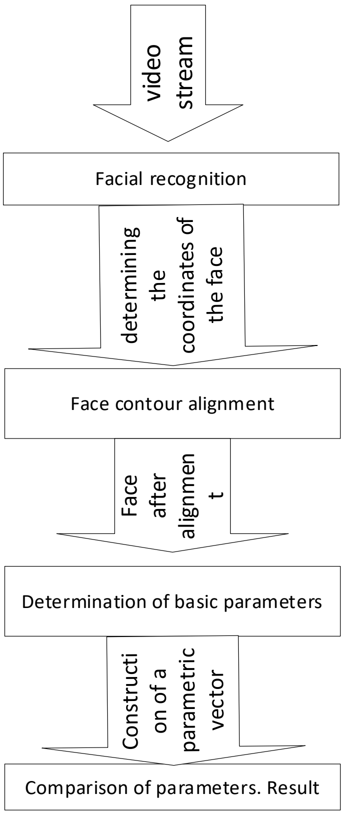
video stream
Facial recognition
determining the coordinates of the face
Face contour alignment
Face after alignmen t
Determination of basic parameters
Constructi on of a parametric vector
Comparison of parameters. Result

Figure 2: Face recognition algorithm

Object detection in offline video. This approach estimates the behavior of perceived objects and works best as a complement to other offline video-based object detection systems [12]. In recent years, various other video object detection systems have emerged that have tried to use 3D convolutional networks that analyze many images simultaneously.

Knowledge-based methods use information about the face, its features, shape, texture, or skin color. In these methods, a certain set of rules is distinguished that a frame fragment must meet to be considered a human face. It is quite easy to define such a set of rules (Fig. 3). All rules are formalized knowledge that a person uses to determine whether a face is a face or not.

For example, the basic rules are: the areas of the eyes, nose, and mouth differ in brightness from the rest of the face; the eyes on the face are always symmetrically positioned relative to each other. Based on these and other similar properties, algorithms are built that check whether these rules are fulfilled in the image during execution. The same group of methods includes a more general method—the pattern-matching method. In this method, a face standard (template) is determined by describing the properties of individual face areas and their specified relative position, with which the input image is subsequently compared.

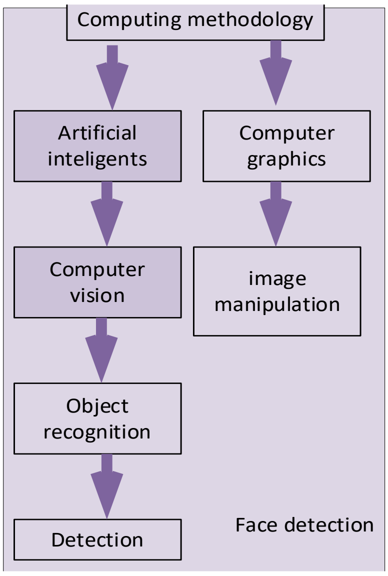

Figure 3: Classification of face detection

Face detection using such methods is performed [13] by searching all rectangular fragments of the image to determine which class the image belongs to.

Viola-Jones object detection [14]. The method was proposed by Paul Viola and Michael Jones and became the first method to demonstrate high results in real-time image processing. The method has many implementations, including as part of the OpenCV computer vision library (cvHaarDetectObjects function). The advantages of this method are high speed (due to the use of a cascade classifier); high accuracy in detecting turned faces at an angle of up to 30 degrees. The disadvantages include a long training time. The algorithm needs to analyze a large number of test images.

The method of comparison on graphs (Elastic graph matching) [15]. This method is related to 2D modeling. Its essence lies in the comparison of graphs describing faces (a face is represented as a grid with an individual location of vertices and edges). Faces are represented as graphs with weighted vertices and edges. At the recognition stage, one of the graphs, the reference graph, remains unchanged, while the other is deformed to best match the first graph. In such recognition systems, graphs can have a rectangular lattice and a structure formed by characteristic (anthropometric) points of faces.

Graph edges are weighted by the distances [16] between adjacent vertices. The difference (distance, discriminative characteristic) between two graphs is calculated using a certain deformation cost function that takes into account both the difference between the feature values calculated in the vertices and the degree of deformation of the graph edges.

The graph is deformed by shifting each of its vertices by a certain distance in certain directions relative to its original location and choosing such a position at which the difference between the feature values in the vertex of the deformed graph and the corresponding vertex of the reference graph is minimal. This operation is performed in turn for all graph vertices until the smallest total difference between the features of the deformed and reference graphs is achieved. The value of the deformation cost function at this position of the graph will be the measure of the difference between the input face image and the reference graph. This “relaxation” deformation procedure should be performed for all reference faces in the system database. The result of the system’s recognition is the reference with the best value of the deformation cost function.

The disadvantages of the method include the complexity of the recognition algorithm and the complicated procedure for entering new templates into the database.

The best results in face recognition were shown by the CNN or convolutional neural network. The success is due to the ability to understand the two-dimensional topology of the image, unlike the multilayer perceptron.

The distinctive features of CNN are local receptor fields (providing local two-dimensional connectivity of neurons), common weights (providing detection of some features anywhere in the image), and hierarchical organization with spatial subsampling. Thanks to these innovations, the CNN provides partial resistance to scale changes, shifts, rotations, changes in angle, and other distortions.

CNN was developed in DeepFace, which was acquired by Facebook to recognize the faces of its social network users.

Geometric face recognition method [17] is one of the first face recognition methods used. The methods of this type of recognition involve the selection of a set of key points (or areas) of the face and the subsequent formation of a set of features. The key points can include the corners of the eyes, lips, the tip of the nose, the center of the eye, etc. The advantages of this method include the use of inexpensive equipment. The disadvantages are as follows: low statistical reliability, high lighting requirements, and mandatory frontal image of the face, with small deviations. It does not take into account possible changes in facial expressions.

The method of flexible comparison on graphs [18], the essence of which is to compare graphs describing the image of a person's face. Some publications indicate 95–97% recognition efficiency even in the presence of different emotional expressions and changes in the angle when forming a face image up to 15 degrees. However, it takes about 25 seconds to compare the input face image with 87 reference images. Another disadvantage of this approach is the low manufacturability of memorizing new standards, which generally leads to a non-linear dependence of the operating time on the size of the face database. The main advantage is low sensitivity to face illumination and changes in face angle, but this approach itself has lower recognition accuracy than methods built using neural networks.

The Principal Component Method (PCM) [19] reduces the recognition or classification process to the construction of a certain number of principal components of images for an input image. However, in cases where there are significant changes in illumination or facial expression in the face image, the effectiveness of the method is significantly reduced.

The Viola-Jones method [14] allows you to detect objects in images in real-time. The method works well when observing an object at a small angle, up to about 30°. The recognition accuracy using this method partially reaches over 90%, which is a good result. However, at a deviation angle of more than 30°, the recognition probability drops sharply. This feature makes it impossible to detect a face at an arbitrary angle. Use of neural networks.

One of the best results in face recognition is achieved by using CNNs, which are a logical development of such architectures as cognition and recognition. The success is due to the ability to take into account the two-dimensional topology of the image, unlike the multilayer perceptron. Thanks to these innovations, the ANN provides partial resistance to scale changes, shifts, rotations, changes in perspective, and other distortions. Testing of the ANN on the ORL database containing images of faces with slight changes in lighting, scale, spatial rotation, position, and various emotions showed 96% recognition accuracy. The disadvantages of methods based on neural networks include the addition of a new reference face to the database, which requires complete retraining of the network on the entire available set, and this is a rather lengthy procedure that, depending on the size of the sample, requires hours of work or even several days.

Local Binary Patterns (LBPs) [15] were first proposed in 1996 to analyze the texture of halftone images. Studies have shown that LBPs are invariant to small changes in lighting conditions and small image rotations. LBW-based methods work well when using images of faces with different facial expressions, different lighting, and head turns. Among the disadvantages is the need for high-quality image preprocessing due to high sensitivity to noise, as the number of false binary codes increases in its presence.

Hidden Markov models [16]. A hidden Markov model is a statistical model that simulates the operation of a process similar to a Markov process with unknown parameters. According to the model, the task is to find unknown parameters based on other observed parameters. The obtained parameters can be used in further analysis for face recognition. From the point of view of recognition, an image is a two-dimensional discrete signal. The observation vector plays an important role in building an image model. To avoid discrepancies in descriptions, a rectangular window is usually used for recognition. To avoid losing data areas, rectangular windows should overlap each other. The values for overlap, as well as the recognition areas, are selected experimentally. Before use, the model must be trained on a set of pre-labeled images. Each label has its number and defines a characteristic point that the model will have to find when adapting to a new image.

3. Face Detection

MediaPipe Face Detection is a face detection software product that includes 6 landmarks and support for multiple faces. It is based on BlazeFace [17], a lightweight and high-performance face detector specifically designed for mobile GPUs. The detector's real-time performance allows it to be applied to any real-time video stream that requires an accurate face region to be used as input to other task-specific models, such as 3D face keypoint estimation (e.g., MediaPipe Face Mesh), facial features, or facial expression classification, and face region segmentation. BlazeFace utilizes a simplified feature extraction network inspired by MobileNetV1/V2, but distinct from it, a GPU-friendly binding scheme modified from Single Shot MultiBox Detector (SSD).

A collection of detected faces, where each face is represented as a proto-message containing a bounding box and 6 key points (right eye, left eye, nose tip, the center of the mouth, right ear tragion, and left ear tragion). The bounding box consists of xmin and width (both normalized to [0.0, 1.0] by the width of the image), and ymin and height (both normalized to [0.0, 1.0] by the height of the image). Each key point consists of x and y, which are normalized to [0.0, 1.0] by the width and height of the image, respectively (Fig. 4).

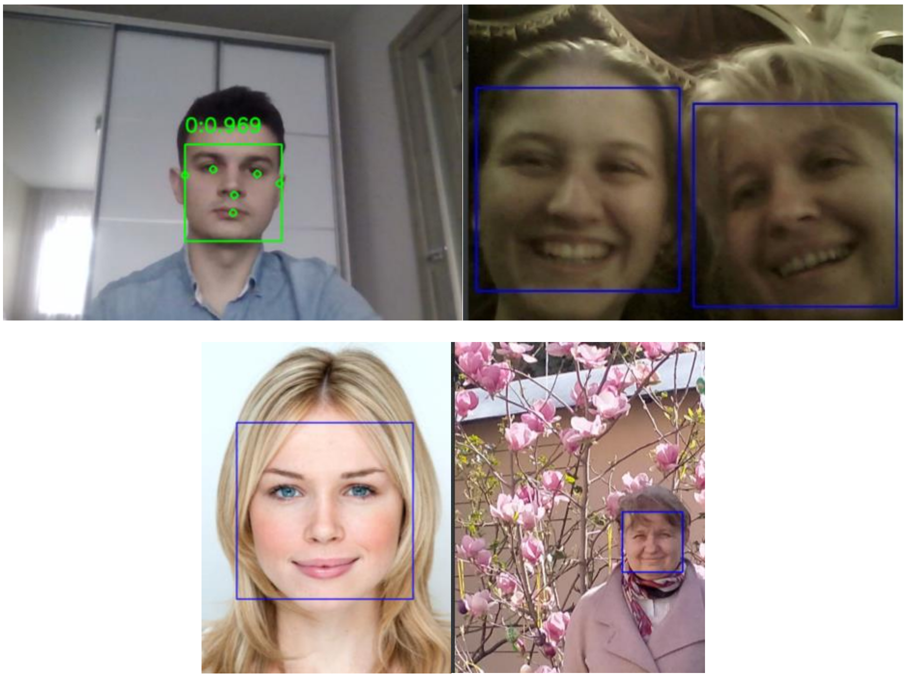

Figure 4: Face capture in video

4. Face Mash

MediaPipe Face Mesh is a solution that estimates 468 3D facial landmarks in real-time, even on mobile devices [18, 19]. The program uses machine learning to determine the 3D surface of the face, requiring only a single camera input without the need for a special depth sensor. Using a simplified modeling architecture along with GPU acceleration throughout the pipeline, the solution delivers real-time performance that is critical.

Additionally, the solution comes with a face transformation module that bridges the gap between facial landmark estimation and useful real-time Augmented Reality (AR) applications [20]. It establishes a metric 3D space and uses the positions of facial landmarks on the screen to estimate facial transformations in that space. The face transformation data consists of conventional 3D primitives, including a face pose transformation matrix and a triangular face mesh [21]. A lightweight statistical analysis method called Procrustes Analysis is used to drive robust, efficient, and portable logic. The analysis is performed on the CPU and has a minimal speed footprint.

The machine learning pipeline consists of two real-time deep neural network models that work together [22]: a detector that works on the full image and calculates the location of the face, and a 3D facial landmark model that works on these locations and predicts an approximate 3D surface using regression. Accurate face cropping significantly reduces the need for conventional data augmentation.

The pipeline is implemented as a MediaPipe graph that uses a face landmark subgraph from the face landmark module and visualizes using a special face renderer subgraph. The face landmark subgraph internally uses the face_detection_subgraph from the face detection module.

The face detector is the same BlazeFace model used in MediaPipe Face Detection.

For 3D facial landmarks, we applied transfer learning and trained the network with multiple objectives: the network simultaneously predicts 3D landmark coordinates on synthetic visualized data and 2D semantic contours on annotated real-world data. The resulting network provided us with reasonable predictions of 3D landmarks not only on synthetic but also on real-world data [23, 24].

The 3D landmark network receives a cropped video frame as input without additional depth input. The model outputs the positions of the 3D points, as well as the probability of the presence and proper alignment of a face in the input data [25, 26]. A common alternative approach is to predict a 2D heat map for each landmark, but it does not lend itself to depth prediction and has high computational costs for so many points. We further improve the accuracy and reliability of our model by iterative loading and refining the predictions. In this way, we can increase our dataset to increasingly complex cases such as grimaces, obliques, and occlusions.

This method can be used for a variety of face masking applications (Fig. 5).

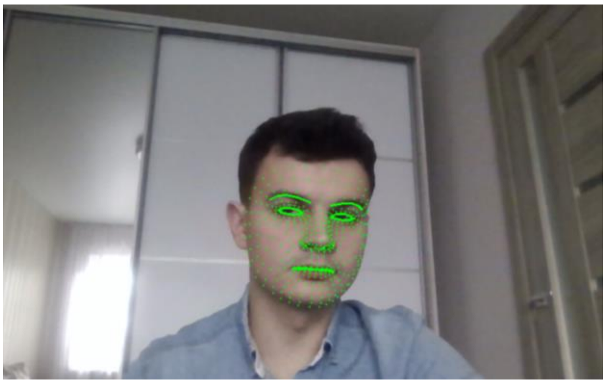

Figure 5: Creating a face mask in a video track

5. Model Development

There are two models in this solution: general and landscape. Both models are based on MobileNetV3 with modifications to make them more efficient. The general model works with a 256×256×3 (HWC) tensor and outputs a 256×256×1 tensor representing the segmentation mask. The landscape model is similar to the general model but works on a 144×256×3 (HWC) tensor. It has fewer FLOPs than the regular model and is therefore faster. MediaPipe Selfie Segmentation automatically resizes the input image to the right tensor size before feeding it to the ML model [27].

The general model also supports ML Kit, and the landscape model option supports Google Meet (Fig. 6).

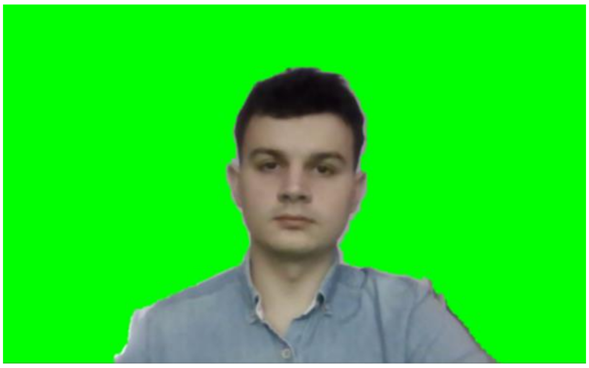

Figure 6: Landscape model—segmentation mask

During this experiment, the issue of recognizing objects in a video stream was considered. The main Python libraries that can be used to recognize and classify objects from video are highlighted. MediaPipe methods for achieving a particular result in recognition are clearly described (Fig. 7).

```
img = cv2.imread(img_path)
cv2.imshow('image', img)
cv2.waitKey(0)
gray = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
faces = face_cascade.detectMultiScale(gray,1.1,5)
faces_detected = "Знайдено обличчя: " + format(len(faces))
print(faces_detected)
for (x, y, w, h) in faces:
    cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2)
    cv2.imshow('img', img)
    cv2.waitKey()
```

Figure 7: Fragment of the face detection program

6. Experimental Results

About 50 images of graduating master’s students were used. The images were taken from mobile phones. Subsequently, after taking photos, they recorded videos in MPEG7 format. In the experiment, the placement of the face to the plane of the photograph was taken into account. The photo was taken in the style of a selfie, a documentary photo, and a portrait. In addition, these images had different variations in quality and contained several facets of variation in color, position, scale, rotation, pose, and facial expression. We present the detection results in Tables 1 and 2 for the HHI MPEG7 image set. The face was fascinated by tracking.

Table 1
FP: False Positives, DR: Detection Rate

Out of man	Frontal	Close to the frontal	Semi-profile	Profile
	Selfi			
Number of images	12	10	7	15
Image size				
FP: False Positives	6204	5205	3090	2580
DR: Detection Rate	87%	85%	90%	99%
Time (sec)	10мс			
	Portrait			
FP: False Positives	5290	5005	2590	2800
DR: Detection Rate	93%	92%	85%	95%
Time (sec)	18 mc			
	Documentary photography			
FP: False Positives	3458	FP: False Positives	3458	FP: False Positives
DR: Detection Rate	85%	DR: Detection Rate	85%	DR: Detection Rate
Time (sec)	22 mc			

7. Acknowledgments

Research on face detection in a video stream has been conducted. This is done using the MediaPipe Face Detection library. The results of face detection are shown in Fig. 4. Frames in the form of photos are recorded in the video stream. The DeepFace library is used to capture faces. If there are several faces in the frame, then DeepFace captures several faces. Experiments were carried out when taking pictures in the genre of the selfie, portrait photography, and documentary photography. The photo was taken from the frontal, close to the frontal and profile. The performed recognition showed a high percentage of face recognition.

8. References

[1] A. Yilmaz, O. Javed, M. Shah, Object Tracking: A Survey. ACM Computing Surveys (CSUR), 38(4) (2006) 13-es.

[2] T. Huang, Computer Vision: Evolution and Promise, CERN School Comput., 1996, 21–25. doi: 10.5170/CERN-1996-008.21

[3] Z. Pang, Z. Li, N. Wang, Simpletrack: Understanding and Rethinking 3D Multi-Object Tracking, ECCV 2022 Workshops: Tel Aviv, Israel, October 2022, 680–696. doi:10.1007/978-3-031-25056-9_43

[4] O. Iosifova, et al., Analysis of Automatic Speech Recognition Methods, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 2923 (2021) 252–257.

[5] K. Khorolska, et al., Application of a Convolutional Neural Network with a Module of Elementary Graphic Primitive

Classifiers in the Problems of Recognition of Drawing Documentation and Transformation of 2D to 3D Models, Journal of Theoretical and Applied Information Technology 100(24) (2022) 7426–7437.

[6] V. Sokolov, P. Skladannyi, A. Platonenko, Video Channel Suppression Method of Unmanned Aerial Vehicles, in: IEEE 41st International Conference on Electronics and Nanotech-nology (2022) 473–477. doi: 10.1109/ELNANO54667.2022.9927105

[7] I. Delibaşoğlu, Moving Object Detection Method with Motion Regions Tracking in Background Subtraction, Signal, Image and Video Processing, (2023) 1–9. doi: 10.1007/s11760-022-02458-y

[8] X. Yu, Evaluation of Training Efficiency of Table Tennis Players Based on Computer Video Processing Technology, Optik, 273 (2023) 170404.

[9] L. Nixon, How Do Destinations Relate to One Another? A Study of Destination Visual Branding on Instagram, ENTER eTourism Conference, 2023, 204–216.

[10] C. Xiao, Z. Luo, Improving Multiple Pedestrian Tracking in Crowded Scenes with Hierarchical Association, Entropy, 25(2) (2023) 380.

[11] S. Garcia, et al., Face-To-Face and Online Teaching Experience on Experimental Animals and Alternative Methods with Nursing Students: A Research Study, BMC Nursing, 22(1) (2023) 1–10.

[12] M. Lee, Y. Chen, Artificial Intelligence Based Object Detection and Tracking for a Small Underwater Robot, Processes, 11(2) (2023) 312.

[13] A. Boyd, et al., CYBORG: Blending Human Saliency Into the Loss Improves Deep Learning-Based Synthetic Face Detection, IEEE/CVF Winter Conference on Applications of Computer Vision, 2023, 6108–6117.

[14] B. Hassan, F. Dawood, Facial Image Detection Based on the Viola-Jones Algorithm for Gender Recognition, Int. J. Nonlinear Analysis Appls. 14(1) (2023) 1593–1599.

[15] E. Hartman, et al., Elastic Shape Analysis of Surfaces with Second-Order Sobolev Metrics: A Comprehensive Numerical Framework, Int. J. Comput. Vision, 2023, 1–27.

[16] E. Rica, S. Álvarez, F. Serratosa, Learning Distances Between Graph Nodes and Edges, Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR Int. Workshops, S+SSPR 2022, Montreal, QC, Canada, August 2022, 103–112.

[17] X. Qi, et al., A Convolutional Neural Network Face Recognition Method Based on BiLSTM and Attention Mechanism, Computational Intelligence and Neuroscience (2023).

[18] Y. Yasuda, et al., Flexibility Chart 2.0: An Accessible Visual Tool to Evaluate Flexibility Resources in Power Systems. Renewable and Sustainable Energy Reviews, 174 (2023) 113116.

[19] G. Ramadan, et al., Impact of PCM type on Photocell Performance Using Heat Pipe-PCM Cooling System: A Numerical Study, J. Energy Systs. 7(1) (2023) 67–88.

[20] S. Sut, et al., Automated Adrenal Gland Disease Classes Using Patch-Based Center Symmetric Local Binary Pattern Technique with CT Images, J. Digital Imaging (2023) 1–14.

[21] R. Glennie, et al., Hidden Markov Models: Pitfalls and Opportunities in Ecology. Methods in Ecology and Evolution, 14(1) (2023) 43–56.

[22] N. Bansal, et al., Real-Time Advanced Computational Intelligence for Deep Fake Video Detection, Appl. Sci. 13(5) (2023) 3095.

[23] B. Deori, D. Thounaojam, A Survey on Moving Object Tracking in Video. Int. J. Inf. Theor. (IJIT), 3(3) (2014) 31–46.

[24] M. Medykovskyy, et al., Methods of Protection Document Formed from Latent Element Located by Fractals, in: X International Scientific and Technical Conference “Computer Sciences and Information Technologies,” 2015, 70–72.

[25] M. Logoyda, et al., Identification of Biometric Images using Latent Elements, CEUR Workshop Proceedings, 2019.

[26] M. Nazarkevych, B. Yavourivskiy, I. Klyuynyk, Editing Raster Images and Digital Rating with Software, The Experience of Designing and Appl. of CAD Systems in Microelectr., 2015, 439–441.

[27] V. Hrytsyk, A. Grondzal, A. Bilenkyj, Augmented Reality for People with Disabilities, in: X Int. Sci. and Technical Conf. “Computer Sciences and Information Technologies,” (2015) 188–191.