Human-machine interactions based on hand gesture recognition using deep learning methods

Lazzat Zholshiyeva[1], Zhanat Manbetova[2], Dinara Kaibassova[3], Akmaral Kassymova[4], Zhuldyz Tashenova[5], Saduakas Baizhumanov[6], Akbota Yerzhanova[7], Kulaisha Aikhynbay[8]

[1]Higher School of Information Technology and Engineering, Astana International University, Astana, Republic of Kazakhstan
[2]Department of Radio Engineering, Electronics and Telecommunications, Faculty of Energy, Saken Seifullin Kazakh Agrotechnical University, Astana, Republic of Kazakhstan
[3]Department of Computer Engineering, Astana IT University, Astana, Republic of Kazakhstan
[4]Department of Information Technology, Zhangir Khan University, Uralsk, Republic of Kazakhstan
[5]Department of Information Security, Eurasian National University named after L.N. Gumilyov, Astana, Republic of Kazakhstan
[6]Faculty of Computer Science, Toraigyrov University, Pavlodar, Republic of Kazakhstan
[7]Department of Technological Machines and Equipment, Faculty of Technology, Saken Seifullin Kazakh Agrotechnical University, Astana, Republic of Kazakhstan
[8]Department of Information Communication Technologies, Information Technology and Energetic Scientific School, Mukhtar Auezov South Kazakhstan University, Shymkent, Republic of Kazakhstan

Article Info

ABSTRACT

Human interaction with computers and other machines is becoming an increasingly important and relevant topic in the modern world. Hand gesture recognition technology is an innovative approach to managing computers and electronic devices that allows users to interact with technology through gestures and hand movements. This article presents deep learning methods that allow you to efficiently process and classify hand gestures and hand gesture recognition technologies for interacting with computers. This paper discusses modern deep learning methods such as convolutional neural networks (CNN) and recurrent neural networks (RNN), which show excellent results in gesture recognition tasks. Next, the development and implementation of a human-machine interaction system based on hand gesture recognition is discussed. System architectures are described, as well as technical and practical aspects of their application. In conclusion, the article summarizes the research results and outlines the prospects for the development of hand gesture recognition technology to improve human-machine interaction. The advantages and limitations of the technology are analyzed, as well as possible areas of its application in the future.

Corresponding Author:

Zhanat Manbetova
Department of Radio Engineering, Electronics and Telecommunications, Faculty of Energy, Saken Seifullin Kazakh Agrotechnical University
010000 Astana, Republic of Kazakhstan
Email: zmanbetova@inbox.ru

1. INTRODUCTION

Human-machine interaction is one of the important aspects of modern information technologies. Hand gesture recognition technologies are an innovative approach to providing a convenient and natural user interaction with computers and electronic devices. One of the most effective and promising methods for hand gesture recognition is the application of deep learning methods. In recent years, deep learning has led to significant breakthroughs in pattern recognition [1], computer vision [2]–[4], and natural language processing

[5]–[8]. These methods allow you to automatically extract high-level features from complex data, which has made them especially attractive for hand gesture recognition problems [9]–[11]. This study is devoted to the research and development of a technology for recognizing human-machine interaction based on hand gesture recognition using deep learning methods. The main goal of this research is to create an efficient and accurate hand gesture recognition system that will allow users to interact with computers and other devices in a natural and intuitive way. To evaluate the performance of the developed models, experiments were carried out on various data sets with real hand gestures. The system's accuracy, speed, and stability are evaluated to determine its effectiveness and applicability in various scenarios.

To evaluate the performance of the developed models, experiments were carried out on various data sets with real hand gestures. The system's accuracy, speed, and stability are evaluated to determine its effectiveness and applicability in various scenarios. The tasks under consideration based on hand gesture recognition using deep learning methods can be diverse, *i.e.,* hand gesture recognition, mouse cursor control, interaction with virtual objects, control of robots, and drones. This research is very relevant and has great potential for application in various areas of human activity, such as in natural interaction, in real world applications, the applicability of deep learning in improving the accuracy and efficiency of the system. In general, hand gesture recognition technology [12]–[15] has significant potential and relevance in the modern world. Its development and application can lead to new innovative products and improved user experience, which makes this topic very important for research and development in the field of computer technology.

Tussupov *et al.* [16] presents a unique learning-based approach to denoising without the use of pure data. The authors have shown that deep neural networks can be trained with a pair of noisy images without the need for pure training data. Lehtinen *et al.* [17] proposes a method for blind defocusing (blur removal) on images using conditional generative adversarial networks (GANs). The authors demonstrated that deep learning with cGAN can successfully recover images with different blur levels. Zhang *et al.* [18] proposed the residual dense network (RDN) for the problem of image super-resolution. RDN combines tight-tie blocks and residual blocks to achieve outstanding results in super-resolution problems. Huang *et al.* [19] presents a method called Noise2Void that allows a deep neural network to be trained to denoise based on a single noisy image without using pure data. This provides an efficient solution for object recognition in noisy images. Zhang and Oh [20] present a deep convolutional network that denoises Monte Carlo tracking images. This method is especially applied in computer graphics, but can be adapted for other problems of object recognition in noisy images.

Zhang *et al.* [21] propose a deep learning model for processing noisy and dark images, which is an important aspect in object recognition in low light conditions. Zhang *et al.* [22] explores the application of networks with deep residual blocks and an attention mechanism to the problem of image super-resolution. Improving image resolution can also be an important step in object recognition in noisy images. Izadi *et al.* [23] presents a deep convolutional network capable of denoising images obtained by the Monte Carlo tracking method. This method is especially used in computer graphics, but it can be adapted to other problems of object recognition in noisy images. Lee and Jeong [24] presents the Noise2Self method, which uses self-surveillance to train a deep neural network on noisy images without the need for clean data. This approach also demonstrates good results in denoising and object recognition problems. Baguer *et al.* [25] presents the deep image prior method, which makes it possible to recover clean images from noisy data without training. This approach is based on the properties of the deep neural network architecture. In study [26], a deep learning model was proposed for processing noisy and dark images, which is an important aspect in object recognition in low light conditions.

2.   METHODS

The use of MediaPipe hand recognition for human-machine interaction based on hand gesture recognition provides new opportunities for creating intuitive, interactive and user-friendly user interfaces, which contributes to the development of more advanced and smart systems. MediaPipe hand recognition is powered by deep neural networks specially trained on huge datasets containing many real-life images and videos with a variety of hand gestures and finger positions. The trained models are used to classify and identify key points on the hand such as fingertips, knuckles, wrist. MediaPipe hand recognition can process both static images and real-time video streams, making it useful for various applications such as hand gesture recognition in interactive systems, virtual reality and augmented reality, mouse cursor and interface control using hand movements, creating collaborative applications where users can draw, write, or interact with content using hand gestures. In the presented work, the MediaPipe hands library was used to detect key points on the hands in the video stream using a webcam. Specifically, the coordinates of the fingertips of the index finger in the image are determined, and then used to control the position of the mouse cursor on the computer screen.

Human-machine interaction based on hand gesture recognition using deep learning methods such as convolutional neural network (CNN) and long short-term memory (LSTM) recurrent neural networks is an exciting and promising area of research and development. The considered CNN and LSTM methods make it possible to create more efficient models for processing video streams with hand gestures and analyzing gesture sequences, which improves the accuracy and reliability of the recognition system. In this context, CNNs are used to process images of hand gestures and extract meaningful features. CNNs are particularly efficient at processing visual information and automatically extract characteristic patterns and structures in images. For successful gesture recognition, CNNs can be used to classify hand images into different categories of gestures.

On the other hand, long short-term memory recurrent neural networks are applied to analyze sequences of hand gestures. LSTMs allow for context and dependencies between successive video frames, making them suitable for analyzing temporal data such as hand gestures. This allows you to create models that can capture the dynamics of gestures and understand the sequence of actions. This approach finds application in various fields such as smart device control, medical applications, virtual and augmented reality, interactive systems, and more. The use of CNNs and LSTMs in hand gesture recognition helps create more accurate and reliable human-machine interaction systems, making the interaction more natural and comfortable for users.

The process of collecting data for training CNN and LSTM models begins with initializing the window and camera, where the parameters for the size of the video stream display window are set and the camera is configured. Hand recognition is then initialized using the MediaPipe library, configuring parameters such as the maximum number of hands to detect and the confidence level. In the main video stream processing loop, each frame is captured, flipped, converted to RGB format and passed to the hand recognition model. If hands are detected, the coordinates of the pointer and thumb are extracted and the distance between them is calculated to determine gestures. Finger pointer coordinates are converted to screen coordinates using the PyAutoGUI library to control the mouse cursor as well as define gesture actions. The data is collected and stored in a comma-separated values (CSV) file "hand_movement_data.csv" for subsequent training of CNN and LSTM models. The current action is displayed on the frame and the process continues until the "q" key is pressed, at which point the camera is released, the windows are closed, and the data is saved.

Thus, the whole process allows you to collect data for training CNN and LTSM models for hand gesture recognition and mouse cursor control based on hand position. Once data is collected, these models can be trained and used to create an interactive application for controlling the mouse cursor using hand gestures. The process of training neural networks based on the above codes includes several stages:

Data preparation: First, the data must be loaded from the "hand_movement_data.csv" file using the Pandas library. The data is then divided into features (X) and labels (y), which are the coordinates of the fingertips and the coordinates of the mouse cursor, respectively. The data is scaled using MinMaxScaler so that feature and label values are in the range 0 to 1.

3. RESULTS AND DISCUSSION

In this paper, the deep learning methods discussed, such as CNN and LSTM, allow the development of efficient gesture recognition systems that can simulate the natural interaction between a person and a computer, which has promising prospects in various fields. One of the main advantages of using CNNs in hand gesture recognition is the ability to automatically extract important features from visual data. CNNs trained on large datasets can recognize hand and gesture features, making them effective tools for classifying various gestures. The ability to use pre-trained CNN models can also significantly reduce training costs and improve system speed.

Using CNN and LSTM recurrent neural networks methods to control the mouse cursor on a computer or device using hand gestures is an interesting and innovative technology that can significantly improve user experience and make interaction with a computer more natural. and convenient. One of the main advantages of using CNNs and LSTMs is their ability to process visual data and analyze sequences of hand movements. CNNs can automatically extract features from hand images, which allows the system to recognize various gestures such as cursor movement, clicks, and scrolling with high accuracy. LSTM, in turn, provides analysis of sequences of gestures. This is especially useful when moving the cursor, as users can perform complex hand movements to accurately position the cursor. LSTM allows the system to take into account the dynamics of movements and adapt to various gestures, which makes cursor control more precise and intuitive. One application for such technology could be the use of hand gestures to control the cursor in virtual or augmented reality. This can provide a convenient and interactive way to interact with virtual objects and environments. However, there are some challenges and limitations that need to be taken into account. First, training efficient CNN and LSTM models requires large amounts of data so that the system

can recognize various gestures with high accuracy. Collecting and labeling such data can be a time-consuming process. Secondly, the system must be able to process real-time gestures for a comfortable user experience. This requires optimization and efficient processing of the video stream with gestures. Consideration should also be given to the system's resilience to varying conditions, such as changing lighting, background noise, or varying hand postures. The system must work reliably and accurately in a variety of scenarios. In general, the use of CNN and LSTM techniques to control the mouse cursor with hand gestures represents a promising technology that can make computer interaction more natural and convenient. This allows users to control the cursor and perform actions on the computer or device more efficiently and intuitively, see Figure 1. However, for the successful implementation of such a system, it is necessary to take into account the challenges associated with training models, processing data in real time and ensuring stable operation in various conditions.
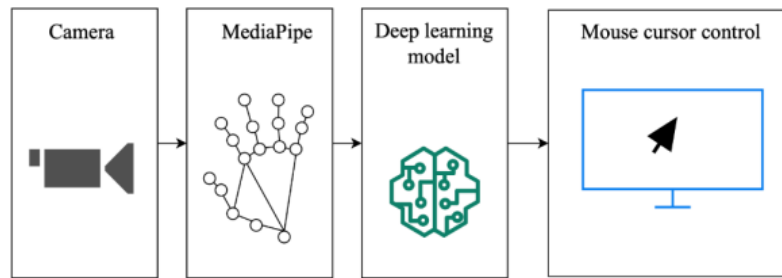
Figure 1. Algorithm for performing human-machine interaction

The prepared data is split into training and test sets using the *train_test_split* function from *sklearn.model_selection*. The training set is used to train the model, and the test set is used to test its performance on new data. A model is created that consists of an LSTM layer with 50 neurons and an output layer with 2 neurons (to predict cursor coordinates). The model is compiled with the Adam optimizer and the *mean_squared_error* loss function. The model is trained on the training set using the fit function. Training takes place over a given number of epochs (100), while the data is divided into batches of 32 in size for more efficient training. After training the model, its performance on the test set is evaluated using the evaluate function. This code calculates the value of the loss function mean squared error (MSE) and the mean absolute error (MAE), as shown in Figure 2.

The trained model is saved to the "hand_movement_lstm_model.h5" file using the save function. Finally, a graph is plotted for the change in the loss function and the mean absolute error on the training and test samples during the training process. As a result of executing the code, we will get a trained LSTM model that can predict the coordinates of the mouse cursor based on the coordinates of the fingertips. If the accuracy of the model on the test set satisfies the requirements, it can be used to control the cursor using hand gestures in real time as shown in Figure 3.
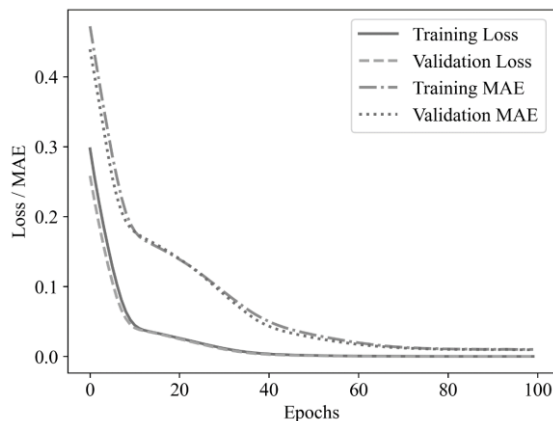

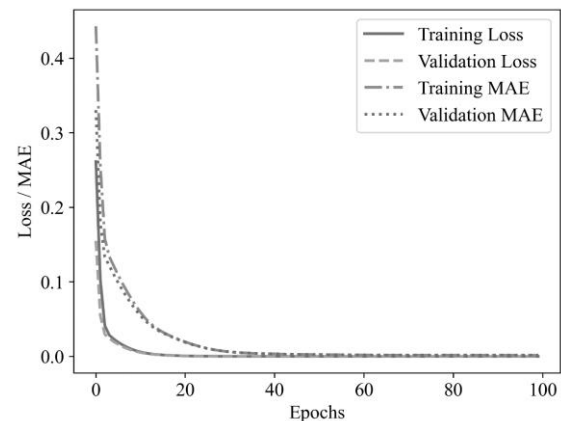Figure 2. LTSM model learning evaluation

Figure 3. CNN model training estimation

As a result of training the considered CNN and LTSM models, it was determined that the CNN does a good job of extracting features from images, which allows you to accurately determine the coordinates of the finger pointer on the video stream. Figure 4 shows the result of an experiment using the CNN model to select a folder on the desktop using hand gestures. The model successfully recognized the hand gestures associated with highlighting a folder on the screen. This includes correctly determining the start and end points of the selection, as well as determining the shape and boundaries of the folder.

Figure 5 shows the result of how a user can use to click on a folder with a squeezing hand gesture. In the future, the model can be successfully applied in real use cases, such as managing files and folders on a computer using hand gestures. When training, it is important to ensure that the model is trained well and tested thoroughly to achieve the desired results and ensure high performance of the system.
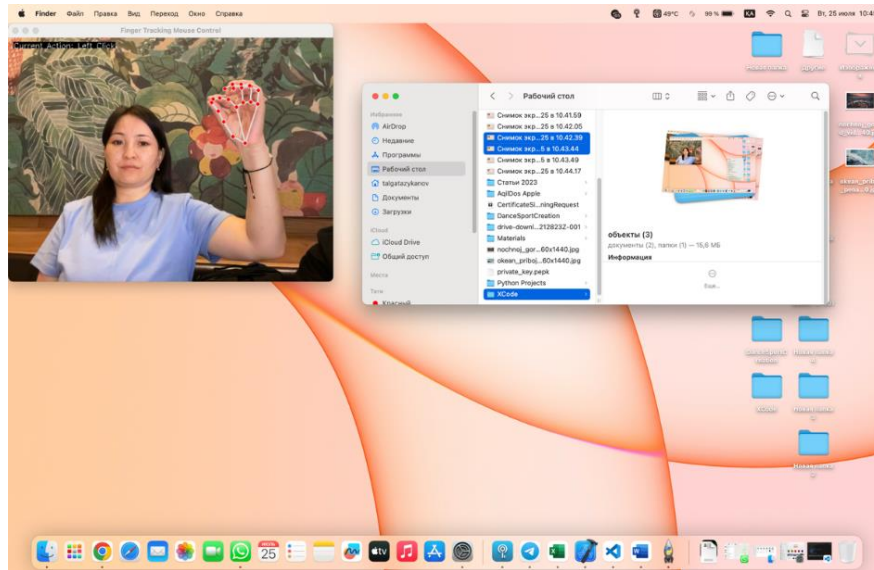
Figure 4. Result of the CNN model for highlighting a folder on the desktop using hand gestures
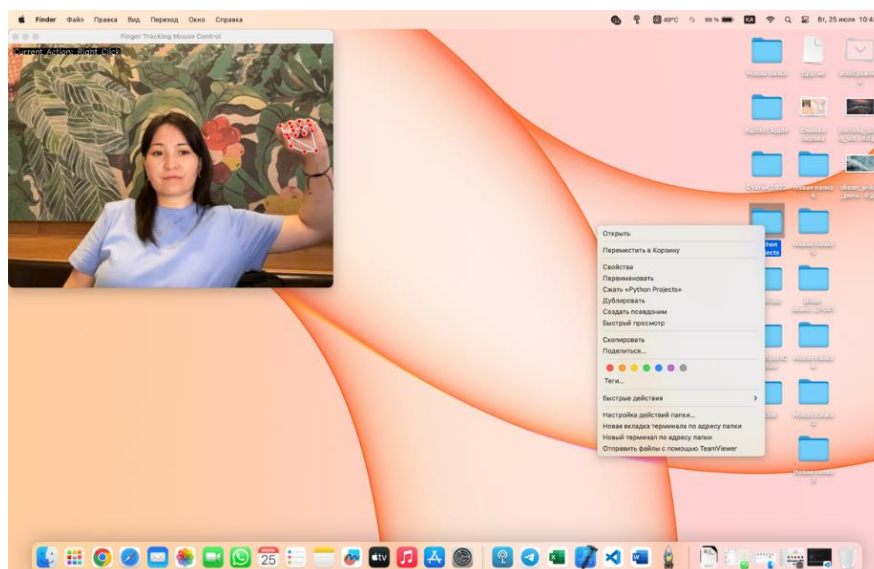
Figure 5. Result of the CNN model for clicking on a folder on the desktop using hand gestures

The result of this research can be successfully applied in an interactive system and device, in virtual and augmented reality, in robot control, in medical and rehabilitation applications, in education and training, in games and entertainment. These are just some of the many areas where hand gesture recognition

technologies using deep learning methods can be successfully applied. With the development of deep learning and computer vision technologies, the capabilities and efficiency of these systems are expected to expand in the future.

4.   CONCLUSION

Using CNN and LSTM recurrent neural networks to control the mouse cursor on a computer or device using hand gestures has improved human-machine interaction, making it more natural and user-friendly. The advantages of using CNNs and LSTMs in this context lie in their ability to efficiently process visual data and analyze hand motion sequences. CNNs automatically extract important features from hand images, which allows the system to recognize various gestures with high accuracy. LSTM, in turn, made it possible to take into account the dynamics of movements and improve the accuracy of cursor control when performing complex gestures.

Both architectures have their advantages and limitations, and the choice of the appropriate model may depend on the specific needs and characteristics of the system. For the task of controlling the mouse cursor with hand gestures, using the CNN model, advantages were identified, such as the ability to process visual data such as images or videos, the ability to automatically extract important features from hand images, which allows recognition of various gestures with high accuracy, and also suitability for classifying and recognizing objects, which is ideal for identifying various hand gestures. The limitation of this model is that it does not take into account sequences of gestures, which can be important for some cursor control scenarios, and it also required more training data, since CNN has many parameters, and large amounts of diverse data are required for successful training.

REFERENCES

[1]   G. Abdikerimova *et al.*, "Applying textural Law's masks to images using machine learning," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 5, pp. 5569–5575, Oct. 2023, doi: 10.11591/ijece.v13i5.pp5569-5575.

[2]   M. H. Guo *et al.*, "Attention mechanisms in computer vision: A survey," *Computational Visual Media*, vol. 8, no. 3, pp. 331–368, Mar. 2022, doi: 10.1007/s41095-022-0271-y.

[3]   A. Parvaiz, M. A. Khalid, R. Zafar, H. Ameer, M. Ali, and M. M. Fraz, "Vision transformers in medical computer vision—A contemplative retrospection," *Engineering Applications of Artificial Intelligence*, vol. 122, Art. no. 106126, Jun. 2023, doi: 10.1016/j.engappai.2023.106126.

[4]   M. Dehghani, A. Gritsenko, A. Arnab, M. Minderer, and Y. Tay, "SCENIC: A JAX library for computer vision research and beyond," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2022, vol. 2022-June, pp. 21361–21366. doi: 10.1109/CVPR52688.2022.02070.

[5]   A. R. Martinez, "Natural language processing," in *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 3, Springer India, 2010, pp. 352–357. doi: 10.1002/wics.76.

[6]   T. Zhang, A. M. Schoene, S. Ji, and S. Ananiadou, "Natural language processing applied to mental illness detection: a narrative review," *NPJ Digital Medicine*, vol. 5, no. 1, Apr. 2022, doi: 10.1038/s41746-022-00589-7.

[7]   C. Wu *et al.*, "Natural language processing for smart construction: Current status and future directions," *Automation in Construction*, vol. 134, Art. no. 104059, Feb. 2022, doi: 10.1016/j.autcon.2021.104059.

[8]   A. Feder *et al.*, "Causal inference in natural language processing: estimation, prediction, interpretation and beyond," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 1138–1158, 2022, doi: 10.1162/tacl_a_00511.

[9]   F. Al Farid *et al.*, "A structured and methodological review on vision-based hand gesture recognition system," *Journal of Imaging*, vol. 8, no. 6, p. 153, May 2022, doi: 10.3390/jimaging8060153.

[10]   Y. Si, S. Chen, M. Li, S. Li, Y. Pei, and X. Guo, "Flexible strain sensors for wearable hand gesture recognition: From devices to systems," *Advanced Intelligent Systems*, vol. 4, no. 2, Jan. 2022, doi: 10.1002/aisy.202100046.

[11]   R. Tchantchane, H. Zhou, S. Zhang, and G. Alici, "A review of hand gesture recognition systems based on noninvasive wearable sensors," *Advanced Intelligent Systems*, Jul. 2023, doi: 10.1002/aisy.202300207.

[12]   A. Bayegizova *et al.*, "Effectiveness of the use of algorithms and methods of artificial technologies for sign language recognition for people with disabilities," *Eastern-European Journal of Enterprise Technologies*, vol. 4, no. 2–118, pp. 25–31, Aug. 2022, doi: 10.15587/1729-4061.2022.262509.

[13]   E. S. Kim, J. W. Shin, Y. S. Kwon, and B. Y. Park, "EMG-based dynamic hand gesture recognition using edge ai for human–robot interaction," *Electronics (Switzerland)*, vol. 12, no. 7, Mar. 2023, doi: 10.3390/electronics12071541.

[14]   S. Wu, Z. Li, S. Li, Q. Liu, and W. Wu, "An overview of gesture recognition," in *International Conference on Computer Application and Information Security ({ICCAIS} 2022)*, Mar. 2023, doi: 10.1117/12.2671842.

[15]   H. Zhou, D. Wang, Y. Yu, and Z. Zhang, "Research progress of human–computer interaction technology based on gesture recognition," *Electronics (Switzerland)*, vol. 12, no. 13, Jun. 2023, doi: 10.3390/electronics12132805.

[16]   J. Tussupov *et al.*, "Applying machine learning to improve a texture type image," *Eastern-European Journal of Enterprise Technologies*, vol. 2, no. 2–122, pp. 13–18, Apr. 2023, doi: 10.15587/1729-4061.2023.275984.

[17]   J. Lehtinen *et al.*, "Noise2Noise: Learning image restoration without clean data," *arxiv.org/abs/1803.04189*, Mar. 2018.

[18]   Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 2472–2481, doi: 10.1109/CVPR.2018.00262.

[19]   T. Huang, S. Li, X. Jia, H. Lu, and J. Liu, "Neighbor2Neighbor: Self-supervised denoising from single noisy images," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp. 14776–14785. doi: 10.1109/CVPR46437.2021.01454.

[20] B. Zhang and K. Oh, "Indirect illumination with efficient Monte Carlo integration and denoising," *Multimedia Tools and Applications*, vol. 80, no. 7, pp. 10167–10185, Nov. 2021, doi: 10.1007/s11042-020-09884-5.

[21] S. Zhang, Y. Zhang, Z. Jiang, D. Zou, J. Ren, and B. Zhou, "Learning to see in the dark with events," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12363 LNCS, Springer International Publishing, 2020, pp. 666–682. doi: 10.1007/978-3-030-58523-5_39.

[22] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11211 LNCS, Springer International Publishing, 2018. doi: 10.1007/978-3-030-01234-2_18.

[23] S. Izadi, D. Sutton, and G. Hamarneh, "Image denoising in the deep learning era," *Artificial Intelligence Review*, vol. 56, no. 7, pp. 5929–5974, Jul. 2023, doi: 10.1007/s10462-022-10305-2.

[24] K. Lee and W. K. Jeong, "Noise2Kernel: Adaptive Self-supervised blind denoising using a dilated convolutional kernel architecture," *Sensors*, vol. 22, no. 11, Jun. 2022, doi: 10.3390/s22114255.

[25] D. O. Baguer, J. Leuschner, and M. Schmidt, "Computed tomography reconstruction using deep image prior and learned reconstruction methods," *Inverse Problems*, vol. 36, no. 9, Sep. 2020, doi: 10.1088/1361-6420/aba415.

BIOGRAPHIES OF AUTHORS

Lazzat Zholshiyeva in 2003, she graduated from the Taraz State University named after M.Kh. Dulaty with a degree in mathematics and computer sciences. In 2012, she received a Master's degree in mechanical engineering. In 2020, she graduated from the doctoral program Astana International University, specialty 8D06101-Computing and software. Her research interests include computer vision, machine learning. She can be contacted at email: lazzat.zholshiyeva@mail.ru.

Zhanat Manbetova in 1999, she graduated from the Korkyt-Ata Kyzylorda State University with a degree in physics and additional mathematics. In 2014, she graduated with a master's degree in radio engineering, electronics and telecommunications from Kazakh Agrotechnical University named after S. Seifullin. In 2022, she defended her doctoral dissertation in the specialty radio engineering, electronics and telecommunications. From 2021 to the present, she is a Doctor of Philosophy PhD of the Department of Radio Engineering, Electronics and Telecommunications of the Kazakh Agrotechnical University named after S. Seifullin. She is the author of more than 40 works. Her research interests include wireless communications, mobile communication systems, GSM, and mobile systems management, as well as mobile communication technologies. She can be contacted at email: zmanbetova@inbox.ru.

Dinara Kaibassova in 1999 she graduated from the Almaty State University named after Abay with a degree in informatica. In 2020 she defended her dissertation in the specialty 6D070300-Informations systems and received a Ph.D. Currently, she is an acting associate professor at the Department of Computer Engineering of Astana IT University. She is the author of more than 80 scientific papers, including 2 monographs, 5 articles in the Scopus database. Scientific interests-image processing, pattern recognition theory, data mining, natural language processing. She can be contacted at email: dindgin@mail.ru.

Akmaral Kassymova candidate of pedagogical sciences, currently works at Zhangir Khan University at the Department of Information Technology, as a professor. She has more than 39 years of scientific and pedagogical experience and more than 100 scientific papers, including 9 articles in the Scopus database, 20 teaching aids. She can be contacted at email: kasimova_ah@mail.ru.

Zhuldyz Tashenova 🆔 🇬 SC Ⓒ PhD, associated professor. Currently works at the Eurasian National University named after L.N. Gumilyov at the Department of Information Security. She has more than 17 years of scientific and pedagogical experience and more than 55 scientific papers, including 8 articles based on Scopus, 2 manuals and developments. She can be contacted by email: zhuldyz_tm@mail.ru.

Saduakas Baizhumanov 🆔 🇬 SC Ⓒ currently a senior lecturer at the Faculty of Computer Science at Toraigyrov University of Kazakhstan. His research interests include mathematical modeling, modeling of management processes, information systems, gamification. He can be contacted at email: b_saduakas@mail.ru.

Akbota Yerzhanova 🆔 🇬 SC Ⓒ defended her doctor of philosophy (Ph.D.) in the specialty 6D070300-Information systems at the L.N. Gumilyov Eurasian National University in Astana. Currently, she works at the Kazakh Agrotechnical University named after S. Seifullin. She is the author or co-author of more than 15 publications. The Hirsch index is 2. Her research interests include geoinformation technologies, pattern recognition, and artificial intelligence. She can be contacted by email: erjanova_akbota@mail.ru.

Kulaisha Aikhynbay 🆔 🇬 SC Ⓒ In 2008, she graduated from the M. Auezov South Kazakhstan State University with a degree in computer sciences. In 2014, she received a Master's degree in master of sciences in specialty computer sciences. Her research interests include machine learning. She can be contacted by email: kulaisha.aikhynbay@gmail.com.