# RESEARCH AND DEVELOPMENT OF IMAGE IMPROVEMENT TOOLS

DOI: 10.36724/2072-8735-2025-19-4-45-56

**Yuri L. Leokhin,**
*Moscow Technical University of Communications and Informatics, Moscow, Russia,* **y.l.leokhin@mtuci.ru**

**Svetlana S. Dymkova,**
*Moscow Technical University of Communications and Informatics, Moscow, Russia,* **ds@media-publisher.ru**

**Timur D. Fatkhulin,**
*Moscow Technical University of Communications and Informatics, Moscow, Russia,* **t.d.fatkhulin@mtuci.ru**

**Manuscript received** 14 March 2025;
**Accepted** 12 April 2025

***Keywords:*** *image, quality, technology, metric, neural network, architecture, autoencoder, method*

**The article deals with the problem of improving image quality using neural network methods. The goal of the research is to develop and study the effectiveness of software that improves image quality by using neural network technologies. The relevance of the work is due to the fact that the use of neural networks (NN) for the subject area under consideration allows the application of new methods that cannot be implemented by traditional approaches. This makes it possible to increase the efficiency and reduce the cost of image processing. The object of the study is graphic images. The subject of the study is the performance indicators (metrics) of neural network methods used to improve image quality. The paper analyzes traditional and neural network methods for increasing resolution and improving image quality. Among the traditional methods, the methods and algorithms of interpolation and filtering were analyzed, and among the neural network methods, the Super-Resolution, Inpainting and Denoising algorithms were analyzed. The paper also presents a comparative analysis of popular software solutions for increasing image resolution. The advantage of using NN methods is shown, and the relevance of software development is determined. The architectures of autoencoders, generative adversarial, convolutional and diffusion neural networks used to improve image quality are analyzed, their advantages and disadvantages are identified. The choice of NN architecture is substantiated. Criteria for assessing image quality are considered. The use of the Structural similarity index measure (SSIM) and Peak signal-to-noise ratio (PSNR) criteria is substantiated. A set of tools for software development is defined. The architecture of a neural network that will solve the problems of existing software solutions is developed. The analysis of the results of the proposed software operation is carried out and conclusions are made about its effectiveness. The work uses the method of theoretical analysis, the method of comparison, as well as experiment and generalization.**

**Information about authors:**

***Yuri L. Leokhin,*** *Moscow Technical University of Communications and Informatics, Rectorate, Moscow, Vice-Rector for Scientific Work, full professor, Dr. Sc. (Tech.), Moscow, Russia. orcid.org/0000-0003-3321-4497*

***Svetlana S. Dymkova,*** *Moscow Technical University of Communications and Informatics, Candidate Sc. (Tech.), Moscow, Russia. orcid.org/0000-0003-1945-9850*

***Timur D. Fatkhulin,*** *Moscow Technical University of Communications and Informatics, Dpt. of MC and IT, Docent, Candidate Sc. (Tech.), Moscow, Russia. orcid.org/0000-0003-0998-1055*

**Для цитирования:**
*Леохин Ю.Л., Дымкова С.С., Фатхулин Т.Д.* Исследование и разработка инструментальных средств повышения качества изображений // T-Comm: Телекоммуникации и транспорт. 2025. Том 19. №4. С. 45-56.

**For citation:**
Yu.L. Leokhin, S.S. Dymkova, T.D. Fatkhulin, "Research and development of image improvement tools," *T-Comm*, 2025, vol. 19, no. 4, pp. 45-56.

## Introduction

The task of improving the quality of images, photographs, increasing their resolution has been relevant for many decades [1, 17, 21]. Currently, a huge amount of information is stored in image format. They are used not only in everyday life, but also in many professional fields. Thus, road surface images obtained using special optical equipment [26-29], photographs of space, macro and microworld, as well as many other images are required for information analysis, research and obtaining important data [1-7, 30-32]. The correctness of the conclusions made during the analysis directly depends on the quality of the images, their detail, clarity, readability. Often, weather conditions, equipment quality and other factors affect the quality of images, so for their further use it is necessary to process them: remove noise, increase clarity, change contrast and much more. In the field of security, individual frames from surveillance cameras can be used for investigations. High image quality will help to obtain more information that can then be used. In this area, a high response speed is often required, so the time spent on improving the quality of images plays an important role. With the development of technology and the improvement of computing technology, the use of various machine learning (ML) methods, in particular neural networks (NN), has become widespread for improving image quality.

To achieve the previously stated goal, it is necessary to solve a number of important interrelated problems:

• analyze existing algorithms, methods, technologies and software solutions used to improve image quality, identify their advantages and disadvantages;

• determine the architecture of the neural network that will be used in developing software to improve image quality;

• justify the choice of criteria for assessing image quality;

• implement the developed software and analyze its effectiveness.

## Analysis of Traditional Image Improvement Methods

The simplest and most widely used traditional methods of improving image quality are interpolation algorithms. Their essence is that existing data is used to obtain expected values at unknown points (information from the open resource https://www.cambridgeincolour.com/ru/tutorials-ru/image-interpolation.htm). Figure 1 shows the principle of the interpolation method when *scaling an image*.

*The nearest neighbor method* is a basic interpolation algorithm. According to it, the value of the nearest pixel is taken to fill an empty pixel. This method gives fast results, but often the image has uneven object boundaries.

Another common method is *bilinear interpolation*. The interpolated value is a weighted average of four pixels around the unknown. The results of bilinear interpolation are smoother than the nearest neighbor method, but there is a risk of getting a blurry image (information from the open source https://www.cambridgeincolour.com/ru/tutorials-ru/image-interpolation.htm).

These two methods and *bicubic interpolation*, which is an improved version of bilinear interpolation, are fairly simple to implement, but often result in unwanted defects: aliasing, blurring, and edge halos (Fig. 2).

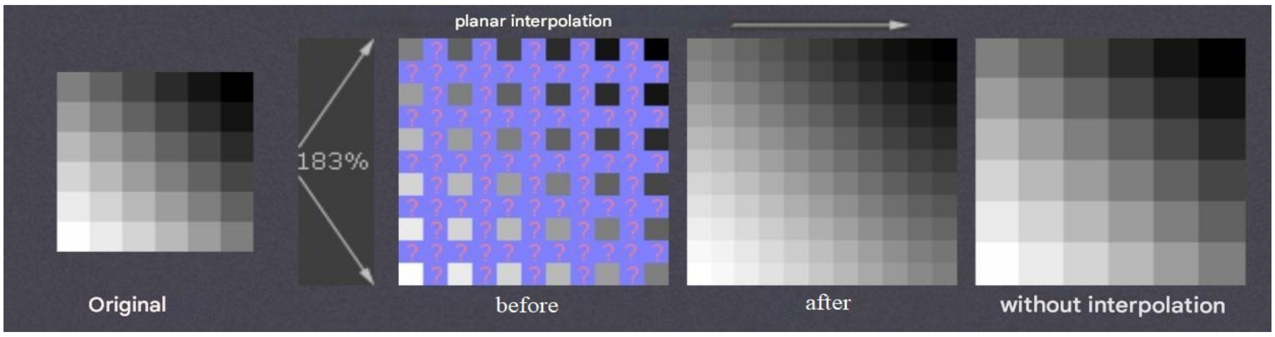


**Fig. 1.** The principle of interpolation when scaling an image

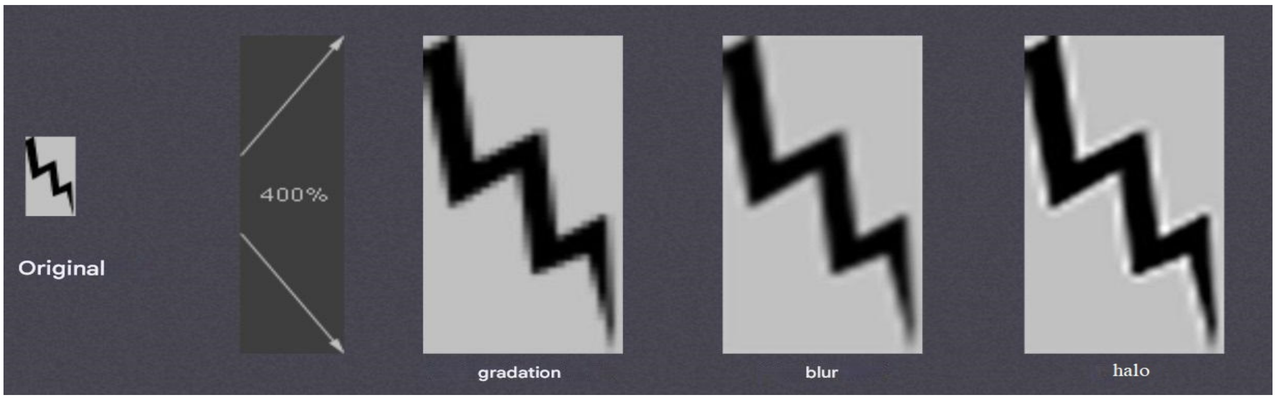


**Fig. 2.** Examples of possible defects when improving image quality

In addition to interpolation algorithms, traditional methods of improving image quality include *filtering*. The principle of filtering is to use various filters to improve clarity and remove digital noise. Filters often introduce artifacts and distortions into the image, and when using filters, smoothing of details and textures may occur [9].

When working with significantly distorted images or images of very low quality, the limitations and disadvantages of traditional methods become most noticeable. They are unable to restore lost details. In addition, pre-definition of interpolation parameters or filters is often required, which makes the use of traditional methods ineffective when working with large amounts of data. To solve these problems, neural network methods for image quality improvement have come to be used.

## Algorithms using neural networks to improve image quality

At the moment, there are a number of algorithms that use neural network technologies to improve image quality. They are selected and applied depending on the task and image quality requirements.

*Super-Resolution* (SR) technology is very popular at the moment and is used in various fields. *Image Super-Resolution* methods can be divided into two groups: classical methods and deep learning-based methods. Early classical methods such as statistical methods, prediction-based methods, fragment-based methods, edge-based methods, and resolved representation methods were often used and the quality improvement with their help was sufficient. However, today, there have been changes in the use of SR methods. With the development of neural network technologies, their application has begun to spread to various tasks, including the task of image quality improvement. Deep learning-based SR methods have shown higher performance than classical methods. Researchers have used various methods ranging from the Convolutional Neural Network (CNN) method to the Generative Adversarial Network (GAN) method [4, 5, 17, 21]. The working principle of the SR algorithm is shown in Figure 3 [10].

The *Inpainting* algorithm is used to fill empty or damaged areas of an image based on existing information. The basic idea is to use neighboring pixels or entire areas to restore missing parts.

There are different methods of inscribing a part of an image, one of them is the option when the prediction based on the global context of the image and the prediction of local textures are optimized. For this, two neural networks are used - one for the context and one for the texture. In this way, it is possible to restore the empty area with greater detail and minimize the appearance of artifacts (Fig. 4) [11].

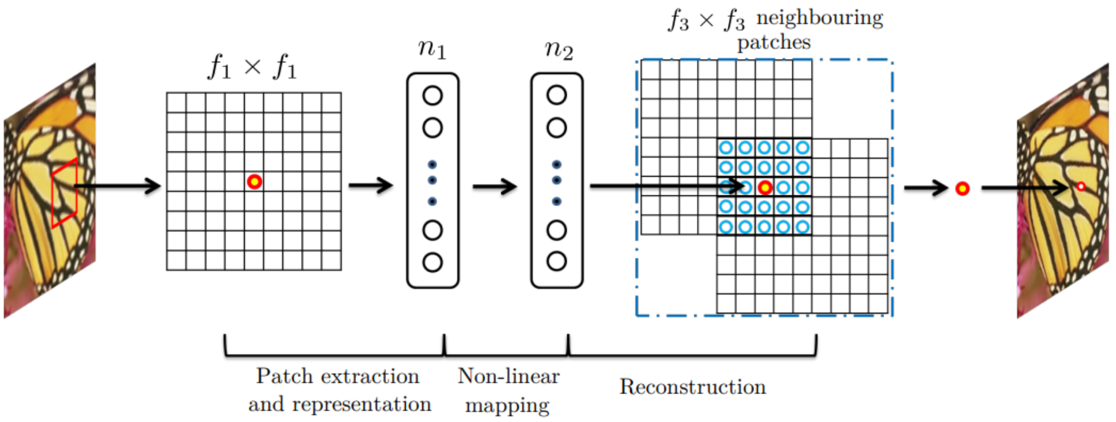


**Fig. 3.** The principle of operation of the SR algorithm [10]

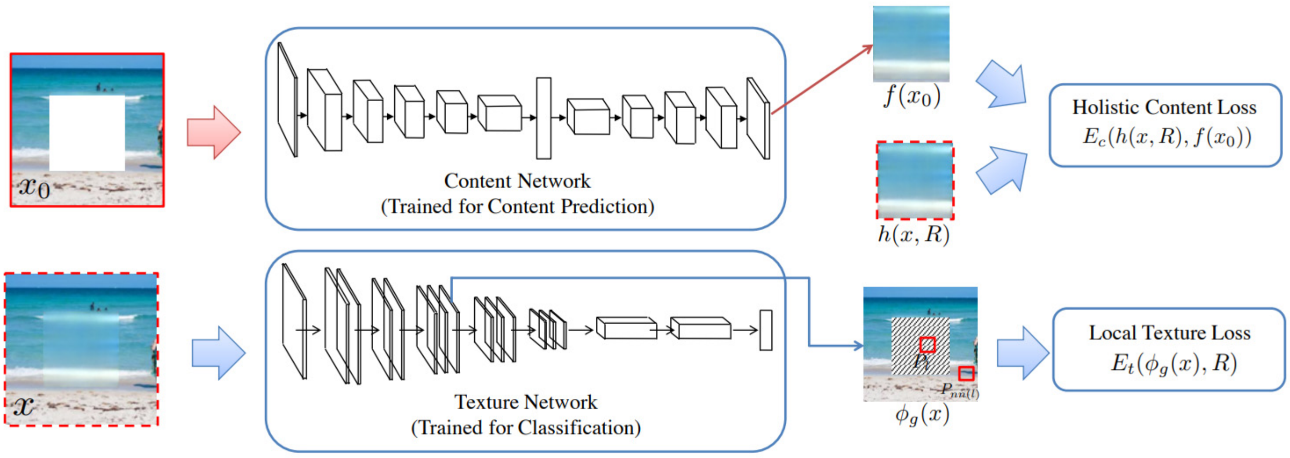


**Fig. 4.** The principle of operation of the Inpainting algorithm [11]

In addition, two directions can also be distinguished: *Deterministic Image Inpainting* and *Stochastic Image Inpainting*. In the first case, as a result of applying the algorithm, one variant of the output value is obtained, while for the second case, it is typical to obtain many variants of output values [8].

The *Denoising* algorithm improves image quality by removing digital noise while preserving details and textures. Initially, there were only traditional noise reduction methods, but now there are methods based on the use of neural networks. The principle of the *Denoising* algorithm is shown in Figure 5 [12].

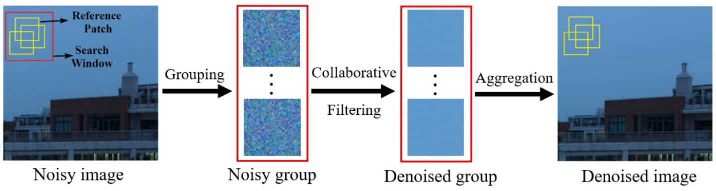


**Fig. 5.** Principle of the Denoising algorithm [12]

It is worth noting that the problem of noise reduction is one of the oldest and most studied problems in image processing, so the existing algorithms for noise reduction show good results. At the moment, solutions for this problem are aimed not only at improving the quality of noise reduction, some of them, for example, diffusion models, have begun to be used for other tasks, such as image generation [10].

### Analysis of existing software solutions for improving image quality

As discussed earlier, image quality improvement is necessary not only in professional fields, but also in everyday life. People need tools to improve images that they can use and that do not require specialized skills and knowledge. Today, there are ready-made tools for improving image quality that have a simple and intuitive design. Such solutions often have other functions, in addition to improving quality, with which you can transform an existing image. For example, it is possible to replace the background, change colors, generate images and many others. Let's consider some popular services that use neural network technologies [11].

The *SnapEdit* service allows you to improve the quality of images, remove objects and restore damaged photos. In addition to the ability to work with photos on the site, users can install the application on Android or iOS. Despite all the advantages of the service, there are also a number of disadvantages. These include a large amount of advertising, the presence of paid functionality and the slowness of the service on devices with insufficient power (information from an open source https://snapedit.app/ru).

The *PicWish* service helps improve the quality of images, remove or replace the background, combine several images, provides the ability to generate an image on request and extract text from an image. The service has an intuitive interface, which makes it easier to work with. The disadvantages of *PicWish* include a limit on the size of the file that will be worked with, as well as the presence of paid content at a high price (information from an open source https://picwish.com/ru/).

The *BigJPG* service increases the size of images and improves their quality. The advantages include working with different image formats and an intuitive interface. But compared to other services, users note a long processing time, as well as periodic blurring of the image as a result of increasing the size (information from an open source https://bigjpg.com/ru).

Based on the analysis of these software solutions, their advantages and disadvantages were identified and presented in Table 1.

Table 1

Advantages and disadvantages of existing image quality improvement services

| Service | Advantages | Disadvantages |
|---|---|---|
| *SnapEdit* | cross-platform;<br>wide functionality | slow service operation;<br>presence of advertising and paid functionality |
| *PicWish* | intuitive interface;<br>wide functionality | file size limit;<br>high subscription price |
| *BigJPG* | work with different image formats;<br>intuitive interface | long processing time;<br>there is a possibility of blurring as a result |

As a result of the analysis of the advantages and disadvantages of existing image quality improvement services, a decision was made to develop software using neural network technologies to improve image quality. This decision is also based on the identified trend in the development of the use of neural networks to solve image processing problems, as well as the need to implement alternative software solutions.

### Analysis of neural network architectures used for image processing

Neural networks are already actively used for various tasks related to image processing. Depending on the specific task, the most suitable architecture is selected. Various neural network architectures already exist to improve image quality, but this area continues to develop. The main types of such NN architectures are autoencoders, convolutional neural networks, generative adversarial networks, and diffusion neural networks [3, 17, 21].

*Autoencoders* are neural networks that learn to reconstruct input data as output. They can be used to remove noise from an image, restore details, and improve clarity [22-25]. The main components of autoencoders are an encoder and a decoder. *The encoder* converts the input image into a latent representation (latent-space). This component consists of several layers, each of which gradually reduces the dimensionality of the data and extracts important features from the image. *The decoder* receives the latent representation from the encoder as input and reconstructs the image. It also has several layers that gradually increase the dimensionality of the data and generate a reconstructed image.

Autoencoders have a number of advantages and disadvantages. Autoencoders do not require labeled data to be trained, which is an important point in some tasks. In addition, during training, autoencoders extract the most important features from the data, thereby reducing its amount. However, this type of neural networks also has its limitations. Such NNs are prone to overtraining, which can affect the results of the model with data that were not displayed in the training set. To prevent this from happening, regularization methods should be carefully selected when working with autoencoders. In addition, this type of neural networks is very sensitive to the choice of hyperparameters, which directly affects the efficiency of the model [2, 6-7].

*Convolutional neural networks (CNN)* – this class of neural networks is specifically designed for image processing. They are often used to improve quality by automatically extracting features from input data [13-16]. The main components of convolutional neural networks are convolutional layers, pooling layers, fully connected layers, and activation layers.

Because this type of neural network identifies the most noticeable features in an image, such as angles and shapes, they show high results in the tasks mentioned above. The disadvantages of convolutional neural networks include such problems as the blurriness of the restored images, as well as the smoothness of the textures on them. This type of neural network is not always able to achieve the same detail as the reference image when restoring image quality, this is influenced by several factors, such as kernel mismatch or overtraining [13].

*Generative Adversarial Networks (GANs)* are a class of neural networks that include two main components: a generator and a discriminator [15]. *The generator* is designed to produce images that are as similar as possible to real images from a training dataset. It consists of several layers, including fully connected and convolutional layers, which help extract features and create high-quality images. The generator takes random noise as input, usually in the form of a vector [18]. *The discriminator* is designed to determine whether the input image is created by the generator (fake) or real. It also consists of several layers that help it analyze images and make decisions about their authenticity. The discriminator is a binary classifier. The generator's goal is to deceive the discriminator, and the discriminator's goal is to more accurately determine the authenticity of images [19].

Generative adversarial networks are most often used to generate images [20-25]. The realism of the images they create is very high in terms of human perception. In addition, this type of neural networks is used to change the style of an image, as well as for other tasks, one of which is improving the quality of images. But despite their great popularity, generative adversarial networks have a number of disadvantages. One of these is the instability of model training. This is due to the fact that this type of neural networks is very sensitive to the settings of the generator and discriminator [22]. If they are not balanced, this leads to the fact that the model will overtrain. With a stronger generator, the discriminator will start making mistakes too quickly and accept the generated images as real. With a stronger discriminator, the generator will start overtraining, trying to deceive the discriminator. Another important disadvantage is that when generating images of higher quality, the model creates incorrect textures. Previously, this was more pronounced, as GANs could not reconstruct small text, generating instead something similar. Now, improvements have been achieved in this problem, but nevertheless, this type of neural networks still generates some textures incorrectly. This can be critical for certain images, for example, if the improved image is then used to analyze the relief and in other cases too [13].

*Diffusion neural networks* – this architecture can be used to generate images and also improve them [17, 21]. The main principle of this architecture is the diffusion method, consisting of two processes: forward diffusion and backward diffusion. *Forward Diffusion* – transforms the input data by gradually adding noise (usually Gaussian noise) to it. *Backward Diffusion* – its purpose is to iteratively restore data from a noisy image, i.e. remove noise to obtain a high-resolution image. Additional data, such as a text description, can be used in this process [14].

Despite the fact that diffusion models create realistic images, including due to the additional data input, they have several significant drawbacks. Due to the fact that the diffusion process is computationally expensive, models can change the original colors when improving an image. This happens due to the small size of batches or insufficient length of the model training process, caused by the limitations of the equipment used to train diffusion neural networks. Another drawback is that diffusion models can show good results on one dataset, but not be as effective for other datasets. This is due to the fact that datasets differ from each other in many image criteria: not only the degree of image distortion (resolution and noise level) are important, but also the variety of objects in the image, as well as the color ratio. When analyzing various neural network architectures for image quality improvement, the advantages and disadvantages of each were highlighted, presented in Table 2.

Table 2

Advantages and disadvantages of CNN, GAN, autoencoders and diffusion neural networks

| Neural network architecture | Advantages | Disadvantages |
|---|---|---|
| *Convolutional Neural Network* | identify the most noticeable characteristics; show good results in classification and detection | blurriness; texture smoothing |
| *Generative Adversarial Network* | creation of realistic images; suitable for various tasks (synthesis, style transfer, etc.) | incorrect texture generation; sensitivity to generator and discriminator settings |
| *Autoencoder* | does not require labeled training data; uses a small number of features | high sensitivity to hyperparameters; not suitable for all datasets |
| *Diffusion neural networks* | additional input; creation of realistic images | change colors; not suitable for all data sets |

After analyzing the advantages and disadvantages of four architectures, it was determined that the autoencoder is the most suitable for the implementation of the software being developed. This architecture will be implemented in further development.

### Definition of criteria (metrics) used to evaluate image quality improvement

When assessing the quality of neural networks for image improvement, it is necessary to use image quality criteria, i.e. metrics by which different models and methods can be compared. Quality criteria are based on various parameters, such as: *sharpness, resolution, contrast, noise, brightness, saturation,* and others. Quality criteria can evaluate an image not only from the point of view of human perception of information, but also track distortions that are imperceptible to the human eye. This may be important if, after image quality has been improved, information from it is read using software, the result of which may be affected by these distortions [15].

It is worth noting that different criteria are used for different tasks and emphasis is placed on some of them depending on which image parameter is most important for a particular task. Let's consider some of the most common criteria.

*Structural similarity index measure (SSIM)* – this quality criterion shows the structural similarity between the improved and original images. The texture of objects in the image, their shape, as well as brightness and contrast are taken into account. It is based on the idea that pixels located nearby have dependencies that contain information about the structure of objects in the image. Images that are more similar to each other will receive a higher value for this criterion than those with worse structural similarity [16, 17].

*Peak signal-to-noise ratio (PSNR)* – this criterion evaluates the noise level in an image by comparing the original image and the improved one and measuring the difference between them. To calculate this criterion, the maximum signal (pixel) value and mean squared error (MSE) are used. A higher value of this criterion indicates better image quality, the lower the value of the indicator, the worse the image quality. There are also several ways to calculate this criterion to evaluate the noise level in a video, and not in a single image [18, 19].

*Perceptual index (PI)* is a quality criterion based on the assessment of a person's visual perception of an image. It takes into account not only physical characteristics, but also how well a person is able to perceive and evaluate an image. A lower value of this criterion indicates that the image is of the highest quality for human perception [20].

*Visual information fidelity (VIF)* – this criterion measures the amount of information transmitted by an image, that is, in addition to taking into account the main characteristics of the image, it also takes into account human perception. This criterion is based on the structure of the human visual system and how a person perceives an image, taking into account that the visual system only allows part of the information through. This criterion is rarely used for images in medicine and some other areas precisely because of this [18].

After analyzing these criteria, it was decided to use the SSIM and PSNR metrics to compare the image quality improvement results of different models and evaluate the effectiveness of the developed software, since the combination of these metrics covers such image imperfections as noise, texture distortion, and changes in the shape of objects. In addition, these two criteria are the most frequently used in research and development of SR solutions, with their help, different architectures are compared, and the effectiveness of the architecture is studied with different degrees of image deformation and with different hyperparameters of the model.

It is important to note that using these two metrics together helps to achieve the highest quality assessment, since PSNR, despite its widespread use due to the simplicity of calculations, has limitations related to the fact that this criterion cannot assess whether the image will be of sufficient quality from the point of view of human perception. SSIM is better at assessing images from a human perspective and also notes similarities in structures and textures, which is lacking when assessing solely by PSNR. However, SSIM is more computationally expensive. In addition, it is worth noting that for such indicators as PSNR and SSIM, diffusion models are inferior to other NNs, although from the point of view of human visual perception, images show high results, which was taken into account when choosing the architecture of the developed NN.

## Autoencoder Architecture Design and Autoencoder Training

Previously, it was decided to select an autoencoder as the architecture for the software being developed to improve image quality using neural network technologies. For further development, it was necessary to determine specific layers, their number, order, and parameters.

During the design, several variants of models were created with different sets of layers and different parameters for these layers. After training these models, their efficiency was checked and, depending on the results, decisions were made to add or remove layers, as well as to change some specific layer arguments. The values of the quality criteria were taken into account, as well as the range in which the models could improve images, since it is important that the software being developed could be used for images distorted to varying degrees.

In total, five types of layers were used in the final model: "Conv2D", "Dropout", "MaxPool2D", "Add" and "UpSampling2D". Convolutional layers were used in both the encoder and decoder (the activation function was "ReLU"), "Dropout" and "MaxPool2D" were used only for the encoder, and "Add" and "UpSampling2D" were used only for the decoder. The high-level Application Programming Interface (API) "Keras" was used to implement the model layers and train it, which made it easier to work with "TensorFlow".

During the implementation, it was found that a regularization coefficient of $10^{-10}$ is optimal for the software being developed, so this coefficient is specified for all convolutional layers of the model. In addition to regularization, a "Dropout" layer was added to the encoder to prevent model overtraining. It excludes neurons from the network with a given probability. In the implemented model, the probability set for the layer is 0.2, since this value showed the best results for training the model. Compared to a model with a lower value, the final model showed better quality metrics, and when compared to a model with a higher probability, it was found that images improved with that model had artifacts. The full architecture of the developed encoder is shown in Figure 6.

Add layers in the decoder architecture were added to share information between the encoder and decoder. This helps the model use more information about the input data for training. The complete architecture of the developed decoder is shown in Figure 7.

The model receives low-resolution images as input. "Adam" was chosen as the optimizer, as it showed the best results during development. The mean square error was set as the loss function.

To train the model, a specialized SR dataset "DIV2K" was selected, containing 800 training images of high and low resolution, 100 test images of high and low resolution, and 100 pairs of "validation" images. For low resolution, several options with different degrees of distortion are presented. This dataset was selected because the images in it have different content: animals, people, landscapes, flowers, etc. Thus, the trained model will be able to further improve the quality of images not only with some limited subject matter, which will be an obvious advantage of the software being developed.

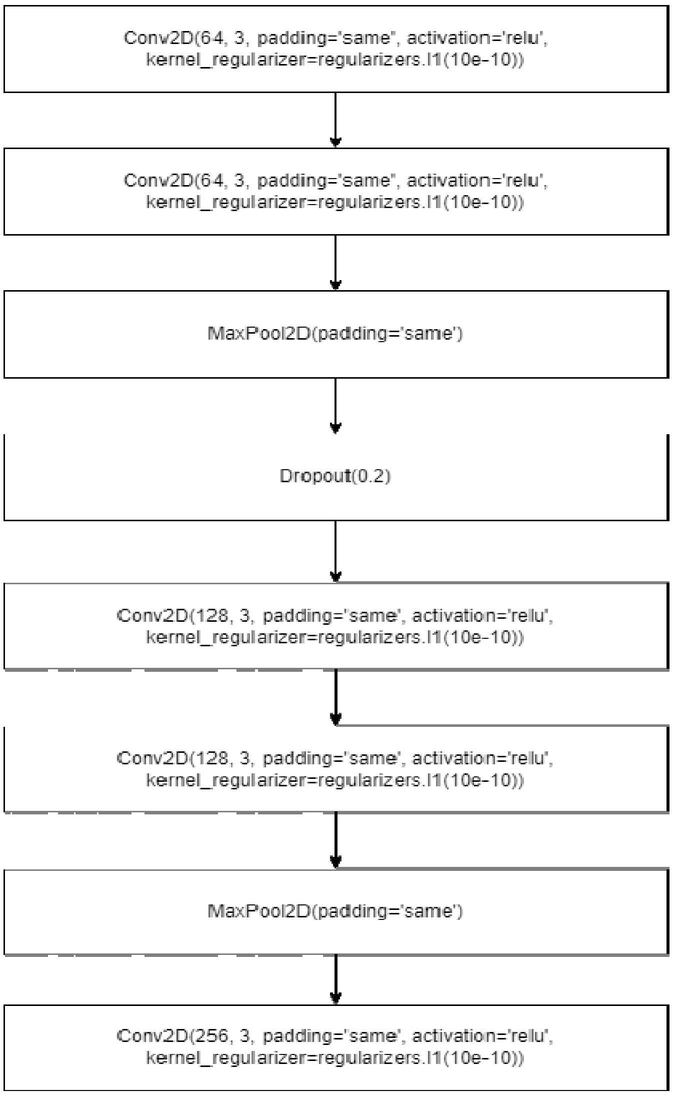


**Fig. 6.** Architecture of the developed encoder

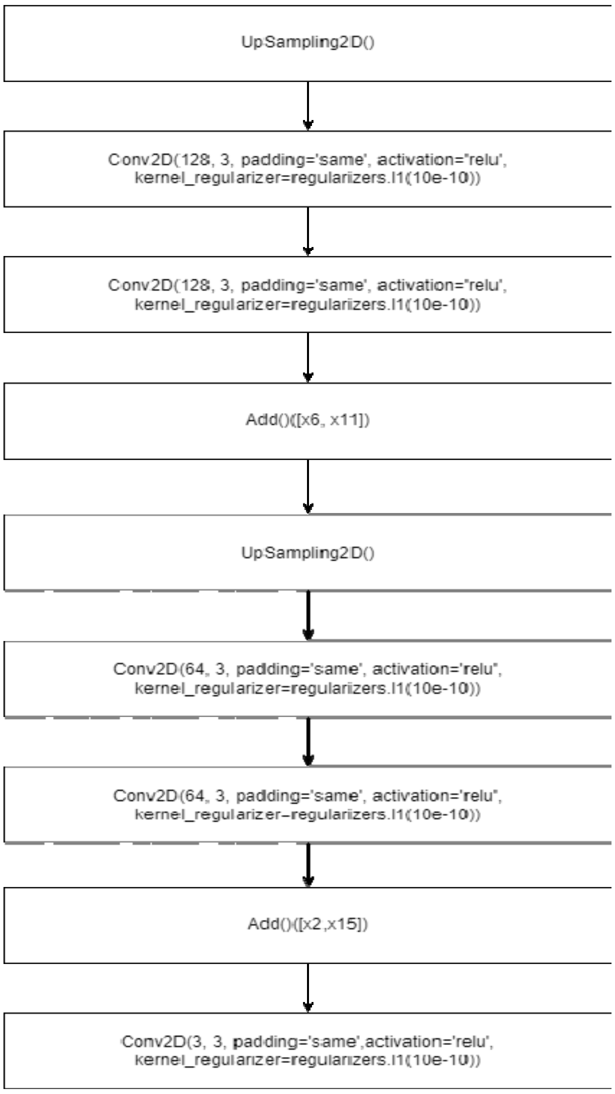


**Fig. 7.** Architecture of the developed decoder

When training models with low-resolution images provided in the dataset, it was noted that the software's efficiency was low. This was due to the fact that the degree of distortion of the images provided was too small, so the model could not cope with more serious damage in the images. In this regard, a function was created that reduces the quality of images according to a specified percentage. This function is used to create low-resolution training and "validation" datasets, as well as to further test the efficiency of the developed software.

The model training used an "EarlyStopping" callback that kept track of the loss function for the validation data, meaning that training would stop when the loss function for the validation data stopped decreasing. It was also specified that the model had to train for at least 50 epochs before training could be stopped. Using this callback also helps prevent the model from overtraining.

The model was trained using Google Colab on a Tesla T4 with 12.7 GB of system RAM and 15 GB of GPU RAM. The initial setting was 200 epochs, but the training was stopped by a callback function after 53 epochs.

The matplotlib library was used to visualize the changes in the loss function and accuracy depending on the epoch, since it is a convenient library that can be used to plot various 2D and 3D graphs, as well as easily adjust their parameters. The figures below show the graph of the loss function for the training and validation data versus the number of epochs on a logarithmic scale (Fig. 8), as well as the dependence of the model accuracy on the training and validation data on the number of epochs (Fig. 9).

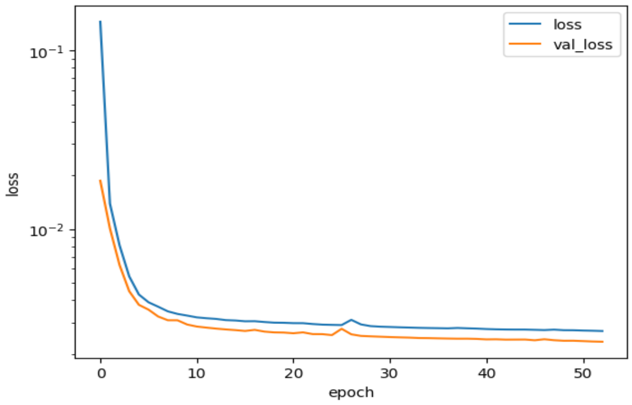


**Fig. 8.** Model loss graph

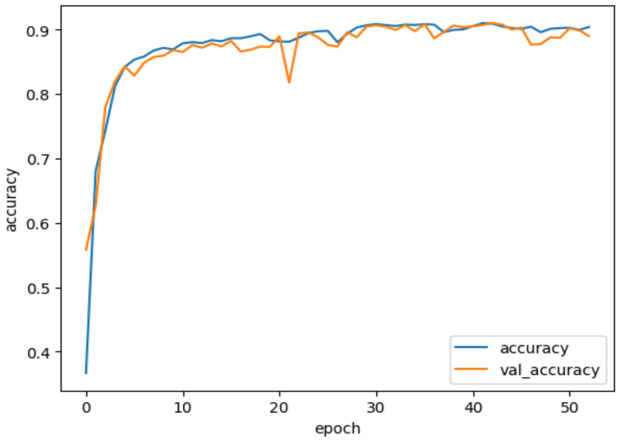


**Fig. 9. Model** accuracy graph

After training was completed, the model was saved to the file "model.h5" for use with the developed graphical interface, as well as for testing its functionality on a local device. This file contains the autoencoder architecture, the weights obtained after training, information about the model compilation and the optimizer.

**Research on the effectiveness of developed software**

To analyze the effectiveness of the developed image quality improvement software, the PSNR and SSIM quality criteria discussed earlier were used. These criteria were calculated between the reference image (high-quality image) and the degraded image, as well as between the reference image and the model output. The degraded images were created using the "bad_image()" function, which was passed the reference image and a percentage starting from 95%. The percentage value was gradually reduced by 5% until the image quality criteria between the original image and the model output became worse than between the original and the degraded image.

PSNR was calculated using the method of the same name in the OpenCV library. The input to this method is a reference image and either a low-quality image or an image improved by the developed software. A higher value of this criterion indicates better image quality. The results of testing the effectiveness of the developed software based on PSNR are presented in Table 3. For convenience, it does not indicate the percentage transferred to the "bad_image()" function, but the percentage of deformation, equal to the difference between 100% and the specified percentage. Also, the values are rounded to two decimal places.

Table 3

PSNR values for the degraded image and the model result

| Percentage of deformation | Original PSNR | PSNR after improvement |
|---|---|---|
| *5* | 73.38 | 75.53 |
| *10* | 72.95 | 74.90 |
| *15* | 72.62 | 74.35 |
| *20* | 72.26 | 73.61 |
| *25* | 71.97 | 72.52 |
| *30* | 71.49 | 72.28 |
| *35* | 71.09 | 71.55 |
| *40* | 70.72 | 70.85 |
| *45* | 70.24 | 70.12 |

The SSIM calculation was performed using "structural_similarity" of the "skimage" package. The input to this method is a reference image and either a low-quality image or an image improved by the developed software. But to calculate the SSIM, the images must first be converted from RGB to "GrayScale". This was implemented using the "OpenCV" library.

As in the case of PSNR, a higher value of this criterion indicates better image quality. The results of testing the effectiveness of the software being developed based on SSIM are presented in Table 4. It also indicates the percentage of image deformation. The values are rounded to two decimal places.

Table 4

SSIM values for degraded image and model output

| Percentage of deformation | Original SSIM | SSIM after improvement |
|---|---|---|
| *5* | 0.88 | 0.94 |
| *10* | 0.87 | 0.93 |
| *15* | 0.86 | 0.92 |
| *20* | 0.85 | 0.91 |
| *25* | 0.84 | 0.88 |
| *30* | 0.82 | 0.87 |
| *35* | 0.80 | 0.85 |
| *40* | 0.79 | 0.82 |
| *45* | 0.76 | 0.78 |
| *50* | 0.76 | 0.71 |

Based on the analysis of the data presented in the tables, it can be noted that the PSNR for the model's performance becomes worse than the original earlier than SSIM. With a deformation percentage of 45%, the original SSIM is less than that obtained as a result of comparing the original and the work of the developed software, which indicates that the structural similarity between the improved image and the original is greater, and accordingly, the quality of the improved image is better.

However, when the image is deformed by 50%, both quality criteria show that the developed software cannot improve the image quality. Moreover, the image quality becomes worse than it was initially. This can be observed if we visualize the noise between the original image and the degraded version and between the original image and the model result.

Let us consider two examples of image quality improvement by the developed software. Figure 10 shows the noise for an image degraded by 50% and for an image improved by the developed software, and Figure 11 shows both images together with the original.

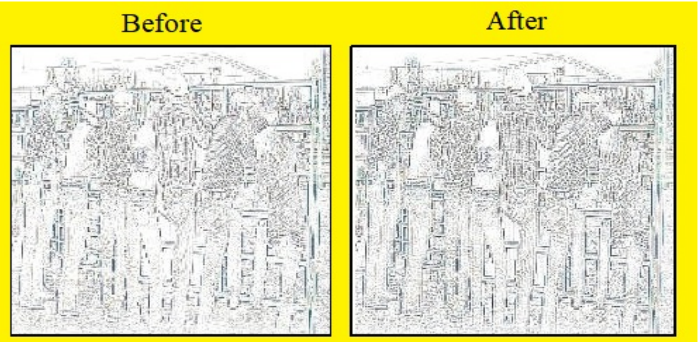


**Fig. 10.** Comparison of noise before and after using the model for the first example


**Fig. 11.** Visualization of all images for the first example

It follows from these figures that with such a degree of deformation of the original image, the model introduces artifacts when trying to improve the image quality.

Let us also consider an example where the model successfully improves the image quality; for this, any deformation value from 5% to 45% can be used, based on the analysis of the effectiveness of the image quality criteria.

Figure 12 shows the noise for an image degraded by 15% and for an image improved using the developed software, and Figure 13 shows both images together with the original.

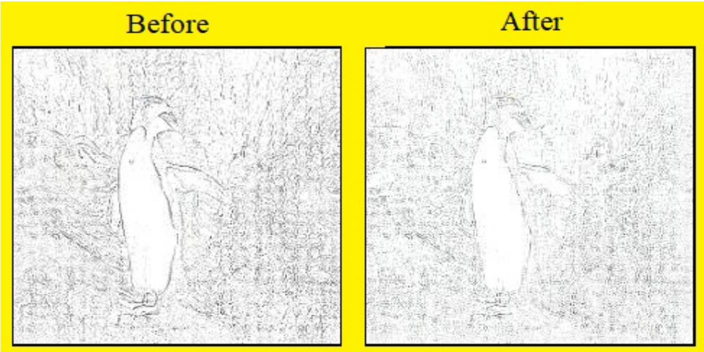


**Fig. 12.** Comparison of noise before and after using the model for the second example


**Fig. 13.** Visualization of all images for the second example

In Figure 10, it can be seen that after improving the image quality using the developed software, the amount of noise has decreased compared to the degraded image, and in Figure 11, it can be seen that the result of the model is clearer compared to the low-resolution image. Also, when analyzing the quality criteria for predicting a heavily deformed image, it was found that the PSNR and SSIM between the original and the result of the model are slightly worse than between the original and the low-resolution image.

## Conclusion

Thus, as a result of the presented study, an analysis of methods and means for improving image quality was conducted, which showed that this task is one of the main ones in computer vision and machine learning today. In addition, when analyzing traditional methods for improving image quality, their limitations were identified that can be circumvented using neural network technologies. An analysis of ready-made solutions for improving image quality based on the use of neural networks showed a number of their shortcomings, as well as the need to implement alternative software solutions.

When comparing different neural network architectures used for the Super Resolution task, their advantages and disadvantages were identified, and a decision was made which architecture to use in the software being developed. In addition, image quality criteria were considered and selected for evaluating the effectiveness of the model, and tools for developing, creating and training the neural network were selected.

After determining the type of neural network, its more detailed architecture was designed, and several models with different parameters were trained and their results were compared to determine the most suitable parameters for the software being developed. A data set was prepared and processed for training, and various methods for regulating model overtraining were used. The results of training and operation of the neural network were visualized and analyzed. This analysis showed the effectiveness and efficiency of the developed software for improving image quality using neural network technologies.

## References

[1] V.M. Vishnevsky, Yu.L. Leokhin, T.D. Fatkhulin, A.V. Zanegin, "Machine learning methods in solving the problem of forecasting demand for specific types of goods," *T-Comm*, vol. 18, no. 10, pp. 34-43. (in Russian)

[2] K.V. Maslov, T.D. Fatkhulin, D.A. Ivanov, "Analysis of business process automation techniques and software development using low-code platforms," *Trudy Severo-Kavkazskogo filiala Moskovskogo tehnicheskogo universiteta svjazi i informatiki*. 2024. No 1. pp. 6-11. (in Russian)

[3] T.D. Fatkhulin, K.V. Boytsov, "Analysis of the functionality of the software used to classify pipes in the enterprise using computer vision methods," *Trudy Severo-Kavkazskogo filiala Moskovskogo tehnicheskogo universiteta svjazi i informatiki*. 2024. No 2. pp. 6-12. (in Russian)

[4] A.A. Myalicheva, T.D. Fatkhulin, "Analysis of machine learning methods for predicting defects in source code," *Trudy Severo-Kavkazskogo filiala Moskovskogo tehnicheskogo universiteta svjazi i informatiki*. 2024. No 2. pp. 16-19. (in Russian)

[5] T.D. Fatkhulin, A.V. Isaev, "Analysis of ARIMA and LSTM models used for cryptocurrency forecasting and investment portfolio determination," *Trudy Severo-Kavkazskogo filiala Moskovskogo tehnicheskogo universiteta svjazi i informatiki*. 2024. No 2. pp. 20-25. (in Russian)

[6] T.D. Fatkhulin, A.V. Isaev, "Efficiency analysis of using ARIMA models for forecasting quotes and determining an investment portfolio in the field of cryptocurrency," *Trudy Severo-Kavkazskogo filiala Moskovskogo tehnicheskogo universiteta svjazi i informatiki*. 2024. No 2. pp. 26-31. (in Russian)

[7] T.D. Fatkhulin, K.V. Boytsov, "Evaluation of the effectiveness of the algorithm based on YOLO v.8 for classifying pipes in an enterprise by photo depending on various conditions," *Trudy Severo-Kavkazskogo filiala Moskovskogo tehnicheskogo universiteta svjazi i informatiki*. 2024. No 2. pp. 51-55. (in Russian)

[8] T.D. Fatkhulin, G.G. Fatkhulina, M.V. Mentus, "Development of a method for forming a prompt to a neural network for the purpose of generating images based on the requirements of computational linguis-

tics," *Trudy Severo-Kavkazskogo filiala Moskovskogo tekhnicheskogo universiteta svyazi i informatiki*. 2023. No. 1. pp. 133-139. (in Russian)

[9] M. Shi, "Fuzzy Image Restoration Algorithm Based on Machine Vision Technology," *2023 International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT)*, Bengaluru, India, 2023, pp. 1-5, DOI: 10.1109/EASCT59475.2023.10392378.

[10] C. Dong, C. C. Loy, K. He and X. Tang, "Image Super-Resolution Using Deep Convolutional Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295-307, 1 Feb. 2016, DOI: 10.1109/TPAMI.2015.2439281.

[11] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang and H. Li, "High-Resolution Image Inpainting Using Multi-scale Neural Patch Synthesis," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 4076-4084, DOI: 10.1109/CVPR.2017.434.

[12] Z. Kong and X. Yang, "Color Image and Multispectral Image Denoising Using Block Diagonal Representation," *IEEE Transactions on Image Processing*, vol. 28, no. 9, pp. 4247-4259, Sept. 2019, DOI: 10.1109/TIP.2019.2907478.

[13] P. Cobelli, S. Nesmachnow and J. Toutouh, "A comparison of Generative Adversarial Networks for image super-resolution," *2022 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*, Montevideo, Uruguay, 2022, pp. 1-6, DOI: 10.1109/LA-CCI54402.2022.9981850.

[14] M. Xu, J. Ma and Y. Zhu, "Dual-Diffusion: Dual Conditional Denoising Diffusion Probabilistic Models for Blind Super-Resolution Reconstruction in RSIs," *IEEE Geoscience and Remote Sensing Letters*, vol. 20, pp. 1-5, 2023, Art no. 6008505, DOI: 10.1109/LGRS.2023.3304418.

[15] L. Yang, Y. Sheng and L. Chai, "A Machine Learning based Reduced-Reference Image Quality Assessment Method for Single-Image Super-Resolution," *2019 Chinese Control Conference (CCC), Guangzhou*, China, 2019, pp. 3571-3576, DOI: 10.23919/ChiCC.2019.8865120.

[16] A. Horé and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," *2010 20th International Conference on Pattern Recognition*, Istanbul, Turkey, 2010, pp. 2366-2369, DOI: 10.1109/ICPR.2010.579.

[17] T. Fatkhulin, Y. Leokhin, M. Mentus, A. Kulikova and R. Alshawi, "Analysis of the Basic Image Generation Methods by Neural Networks," *2023 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED)*, Moscow, Russian Federation, 2023, pp. 1-7, DOI: 10.1109/TIRVED58506.2023.10332668.

[18] M. Martini, "A Simple Relationship Between SSIM and PSNR for DCT-Based Compressed Images and Video: SSIM as Content-Aware PSNR," *2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP)*, Poitiers, France, 2023, pp. 1-5, DOI: 10.1109/MMSP59012.2023.10337706.

[19] O. Keleş, M. A. Yılmaz, A. M. Tekalp, C. Korkmaz and Z. Doğan, "On the Computation of PSNR for a Set of Images or Video," *2021 Picture Coding Symposium (PCS)*, Bristol, United Kingdom, 2021, pp. 1-5, DOI: 10.1109/PCS50896.2021.9477470.

[20] V. R. Ahuja et al., "Siamese-SR: A Siamese Super-Resolution Model for Boosting Resolution of Digital Rock Images for Improved Petrophysical Property Estimation," *IEEE Transactions on Image Processing*, vol. 31, pp. 3479-3493, 2022, doi: 10.1109/TIP.2022.3172211.

[21] Y. Leokhin, T. Fatkhulin and A. Kulikova, "Research of the Influence of the Semantics and Syntax of a Text Request on the Image Quality Generated by Neural Network Techniques," *2023 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED)*, Moscow, Russian Federation, 2023, pp. 1-7, DOI: 10.1109/TIRVED58506.2023.10332723.

[22] E.A. Saksonov, Y.L. Leokhin and V.N. Azarov, "Organization of Information Security in Industrial Internet of Things Systems," *2019 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS)*, Sochi, Russia, 2019, pp. 3-7, DOI: 10.1109/ITQMIS.2019.8928442.

[23] E.L. Portnov, T.D. Fatkhulin, "Technologies aimed at achieving high speed transmission in modern coherent DWDM communication systems," *T-Comm*. 2015. Vol 9. No.8, pp. 34-37. (in Russian)

[24] V.Yu. Deart, T.D. Fatkhulin, "Analysis of current state of transport networks with the purpose of introducing software defined networks (SDN) technology," *T-Comm*, 2017, vol. 11, no.6, pp. 4-9. (in Russian)

[25] V.Yu. Deart, T.D. Fatkhulin, "Analysis of transport software-defined networks (T-SDN) with controlled optical layer to obtain a model providing assesment of the possibility of bandwidth on demand service,". *T-Comm*, 2018, vol. 12, no.4, pp. 35-42. (in Russian)

[26] V.Yu. Deart, T.D. Fatkhulin, "Analysis of the process of creating a superchannel with the necessary capacity in the network organized according to transport software-defined networks (T-SDN) technology," *T-Comm*, 2018, vol. 12, no.10, pp. 23-30. (in Russian)

27. V. Deart and T. Fatkhulin, "Analysis of the functioning of a multi-domain transport software-defined network with controlled optical layer," *2017 21st Conference of Open Innovations Association (FRUCT)*, Helsinki, Finland, 2017, pp. 79-87, DOI: 10.23919/FRUCT.2017.8250168.

[28] Y.L. Leokhin and T.D. Fatkhulin, "Approach to Estimating the Probability of Providing "Cloud" Services in the SDN," *2020 Systems of Signals Generating and Processing in the Field of on Board Communications*, Moscow, Russia, 2020, pp. 1-9, DOI: 10.1109/IEEECONF48371.2020.9078593.

[29] Y.L. Leokhin and T.D. Fatkhulin, "Evaluation of Service Availability in Software-Defined Optical Network," *2021 Systems of Signals Generating and Processing in the Field of on Board Communication*s, Moscow, Russia, 2021, pp. 1-6, DOI: 10.1109/IEEECONF51389.2021.9416122.

[30] A.Y. Kudryashova, "A method of efficient coding of color images under the condition of permissible and forbidden values of color gamut," *T-Comm*. 2019. T. 13. No. 6, pp. 65-70.

[31] S.V. Khadonova, A.V. Ufimtsev and S.S. Dymkova, "Wide application innovative monitoring system with personal smart devices," *2020 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)*, Svetlogorsk, Russia, 2020, pp. 1-5, doi: 10.1109/SYNCHROINFO49631.2020.9166115.

[32] A. Pastukh, V. Tikhvinskiy, S. Dymkova and O. Varlamov, "Challenges of Using the L-Band and S-Band for Direct-to-Cellular Satellite 5G-6G NTN Systems," *Technologies*, vol. 11, no. 4, p. 110, Aug. 2023, doi: 10.3390/technologies11040110.

# ИССЛЕДОВАНИЕ И РАЗРАБОТКА ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ПОВЫШЕНИЯ КАЧЕСТВА ИЗОБРАЖЕНИЙ

**Леохин Юрий Львович,** *Московский технический университет связи и информатики, Москва, Россия,* **y.l.leokhin@mtuci.ru**
**Дымкова Светлана Сергеевна,** *Московский технический университет связи и информатики, Москва, Россия,* **ds@media-publisher.ru**
**Фатхулин Тимур Джалилевич,** *Московский технический университет связи и информатики, Москва, Россия,* **t.d.fatkhulin@mtuci.ru**

**Аннотация**
В работе рассматривается проблема повышения качества изображения нейросетевыми методами. Целью работы является разработка и исследование эффективности программного обеспечения (ПО), которое улучшает качество изображений путем применения нейросетевых технологий. Актуальность работы обусловлена тем, что использование нейронных сетей (НС) для рассматриваемой предметной области позволяет применять новые методы, которые не могут быть осуществлены традиционными подходами. Это дает возможность повысить эффективность и уменьшить стоимость обработки изображений. Объектом исследования являются графические изображения. Предметом исследования являются показатели эффективности (метрики) нейросетевых методов, использующихся для улучшения качества изображений. В работе анализируются традиционные и нейросетевые методы повышения разрешения и улучшения качества изображений. Среди традиционных методов проанализированы методы и алгоритмы интерполяции и фильтрации, а среди нейросетевых - алгоритмы Super-Resolution, Inpainting и Denoising. Также в работе представлен сравнительный анализ популярных программных решений для повышения разрешения изображений. Показано преимущество использования НС методов, а также определена актуальность разработки ПО. Проанализированы архитектуры автоэнкодеров, генеративно-состязательных, сверточных и диффузионных нейронных сетей, применяемых для улучшения качества изображений, выявлены их преимущества и недостатки. Обоснован выбор архитектуры НС. Рассмотрены критерии оценки качества изображения. Обосновано использование критериев Structural similarity index measure (SSIM) и Peak signal-to-noise ratio (PSNR). Определен набор инструментов для разработки программного обеспечения. Разработана архитектура нейронной сети, которая будет решать проблемы существующих программных решений. Проведен анализ результатов работы предлагаемого программного обеспечения и сделаны выводы о его эффективности. В работе применен метод теоретического анализа, методы сопоставления и сравнения, а также эксперимента и обобщения.

***Ключевые слова:*** *изображение, качество, технология, метрика, нейронная сеть, архитектура, автоэнкодер, метод*

**Литература**

1. *Вишневский В.М., Леохин Ю.Л., Фатхулин Т.Д., Занегин А.В.* Методы машинного обучения в решении задачи прогнозирования спроса на отдельные виды товаров // T-Comm: Телекоммуникации и транспорт. 2024. Том 18. №10. С. 34-43.
2. *Маслов К.В., Фатхулин Т.Д., Иванов Д.А.* Анализ технологий автоматизации бизнес-процессов и разработки программного обеспечения с использованием low-code платформ // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. - 2024. - № 1. С. 6-11. EDN HDBOYM.
3. *Фатхулин Т.Д., Бойцов К.В.* Анализ функционала программного обеспечения, применяемого для классификации труб на предприятии методами компьютерного зрения // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2024. № 2. С. 6-12. EDN FVQYQA
4. *Мяличева А.А., Фатхулин Т.Д.* Анализ методов машинного обучения для прогнозирования дефектов в исходном коде // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2024. № 2. С. 16-19. EDN IVJCZF
5. *Фатхулин Т.Д., Исаев А.В.* Анализ моделей ARIMA и LSTM, используемых для прогнозирования криптовалют и определения портфеля инвестиций // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2024. № 2. С. 20-25. EDN ODWOPA
6. *Фатхулин Т.Д., Исаев А.В.* Анализ эффективности использования моделей ARIMA для прогнозирования котировок и определения портфеля инвестиций в области криптовалюты // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2024. № 2. С. 26-31. EDN CESRTK.
7. *Фатхулин Т.Д., Бойцов К.В.* Оценка эффективности алгоритма на основе YOLO v.8 для классификации труб на предприятии по фото в зависимости от различных условий // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2024. № 2. С. 51-55. EDN KWLMYA
8. *Фатхулин Т.Д., Фатхулина Г.Г., Ментус М.В.* Разработка методики формирования запроса к нейросети с целью генерации изображений с учетом рекомендаций компьютерной лингвистики // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2023. № 1. -С. 133-139. EDN PPRTOM
9. *Shi M.* Fuzzy Image Restoration Algorithm Based on Machine Vision Technology// 2023 International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT), Bengaluru, India, 2023, pp. 1-5, DOI: 10.1109/EASCT59475.2023.10392378.
10. *Dong C., Loy C.C., He K., Tang X.* Image Super-Resolution Using Deep Convolutional Networks // IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 2, pp. 295-307, 1 Feb. 2016, DOI: 10.1109/TPAMI.2015.2439281.
11. *Yang C., Lu X., Lin Z., Shechtman E., Wang O., Li H.* High-Resolution Image Inpainting Using Multi-scale Neural Patch Synthesis // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 4076-4084, DOI: 10.1109/CVPR.2017.434.
12. *Kong Z., Yang X.* Color Image and Multispectral Image Denoising Using Block Diagonal Representation // IEEE Transactions on Image Processing, vol. 28, no. 9, pp. 4247-4259, Sept. 2019, DOI: 10.1109/TIP.2019.2907478.

13. *Cobelli P., Nesmachnow S., Toutouh J.* A comparison of Generative Adversarial Networks for image super-resolution // 2022 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Montevideo, Uruguay, 2022, pp. 1-6, DOI: 10.1109/LA-CCI54402.2022.9981850.

14. *Xu M., Ma J., Zhu Y.* Dual-Diffusion: Dual Conditional Denoising Diffusion Probabilistic Models for Blind Super-Resolution Reconstruction in RSIs // IEEE Geoscience and Remote Sensing Letters, vol. 20, pp. 1-5, 2023, Art no. 6008505, DOI: 10.1109/LGRS.2023.3304418.

15. *Yang L., Sheng Y., Chai L.* A Machine Learning based Reduced-Reference Image Quality Assessment Method for Single-Image Super-Resolution // 2019 Chinese Control Conference (CCC), Guangzhou, China, 2019, pp. 3571-3576, DOI: 10.23919/ChiCC.2019.8865120.

16. *Hore A., Ziou D.* Image Quality Metrics: PSNR vs. SSIM // 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 2366-2369, DOI: 10.1109/ICPR.2010.579.

17. *Fatkhulin T. Leokhin Y., Mentus M., Kulikova A., Alshawi R.* Analysis of the Basic Image Generation Methods by Neural Networks // 2023 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED), Moscow, Russian Federation, 2023, pp. 1-7, DOI: 10.1109/TIRVED58506.2023.10332668.

18. *Martini M.* A Simple Relationship Between SSIM and PSNR for DCT-Based Compressed Images and Video: SSIM as Content-Aware PSNR // 2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP), Poitiers, France, 2023, pp. 1-5, DOI: 10.1109/MMSP59012.2023.10337706.

19. *Keles O., Yilmaz M. A., Tekalp A. M., Korkmaz C., Dogan Z.* On the Computation of PSNR for a Set of Images or Video // 2021 Picture Coding Symposium (PCS), Bristol, United Kingdom, 2021, pp. 1-5, DOI: 10.1109/PCS50896.2021.9477470.

20. *Ahuja V.R.* et al., "Siamese-SR: A Siamese Super-Resolution Model for Boosting Resolution of Digital Rock Images for Improved Petrophysical Property Estimation // IEEE Transactions on Image Processing, vol. 31, pp. 3479-3493, 2022, doi: 10.1109/TIP.2022.3172211.

21. *Leokhin Y., Fatkhulin T., Kulikova A.* Research of the Influence of the Semantics and Syntax of a Text Request on the Image Quality Generated by Neural Network Techniques // 2023 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED), Moscow, Russian Federation, 2023, pp. 1-7, DOI: 10.1109/TIRVED58506.2023.10332723.

22. *Saksonov E.A., Leokhin Y.L., Azarov V.N.* Organization of Information Security in Industrial Internet of Things Systems // 2019 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS), Sochi, Russia, 2019, pp. 3-7, DOI: 10.1109/ITQMIS.2019.8928442.

23. *Портнов Э.Л., Фатхулин Т.Д.* Технологии достижения высоких скоростей передачи в современных когерентных DWDM-системах связи // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9, № 8. С. 34-37. EDN ULFNNR

24. *Деарт В.Ю., Фатхулин Т.Д.* Анализ современного состояния транспортных сетей с целью внедрения технологии программно-конфигурируемых сетей (SDN) // T-Comm: Телекоммуникации и транспорт. 2017. Т. 11, № 6. С. 4-9. EDN ZAOKIH

25. *Деарт В.Ю., Фатхулин Т.Д.* Анализ транспортных программно-конфигурируемых сетей (T-SDN) с управляемым оптическим уровнем с целью получения модели, позволяющей оценить возможность предоставления сервиса Bandwidth on Demand // T-Comm: Телекоммуникации и транспорт. - 2018. Т. 12, № 4. С. 35-42. DOI 10.24411/2072-8735-2018-10063. EDN UQHXRY

26. *Деарт В.Ю., Фатхулин Т.Д.* Анализ процесса создания суперканала с необходимой пропускной способностью в сети, построенной по технологии транспортных программно-конфигурируемых сетей (T-SDN) // T-Comm: Телекоммуникации и транспорт. 2018. Т. 12, № 10. С. 23-30. DOI 10.24411/2072-8735-2018-10152. EDN YMWVOH

27. *Deart V., Fatkhulin T.* Analysis of the functioning of a multi-domain transport software-defined network with controlled optical layer // 2017 21st Conference of Open Innovations Association (FRUCT), Helsinki, Finland, 2017, pp. 79-87, DOI: 10.23919/FRUCT.2017.8250168.

28. *Leokhin Y.L., Fatkhulin T.D.* Approach to Estimating the Probability of Providing "Cloud" Services in the SDN // 2020 Systems of Signals Generating and Processing in the Field of on Board Communications, Moscow, Russia, 2020, pp. 1-9, DOI: 10.1109/IEEECONF48371.2020.9078593.

29. *Leokhin Y.L., Fatkhulin T.D.* Evaluation of Service Availability in Software-Defined Optical Network // 2021 Systems of Signals Generating and Processing in the Field of on Board Communications, Moscow, Russia, 2021, pp. 1-6, DOI: 10.1109/IEEECONF51389.2021.9416122.

30. *Kudryashova A.Y.* A method of efficient coding of color images under the condition of permissible and forbidden values of color gamut // T-Comm. 2019. T. 13. № 6. C. 65-70.

31. *Khadonova S.V., Ufimtsev A.V., Dymkova S.S.* Wide application innovative monitoring system with personal smart devices // 2020 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), Svetlogorsk, Russia, 2020, pp. 1-5, doi: 10.1109/SYNCHROINFO49631.2020.9166115.

32. *Pastukh A., Tikhvinskiy V., Dymkova S., Varlamov O.* Challenges of Using the L-Band and S-Band for Direct-to-Cellular Satellite 5G-6G NTN Systems // Technologies, vol. 11, no. 4, p. 110, Aug. 2023, doi: 10.3390/technologies11040110.

***Информация об авторах:***

***Леохин Юрий Львович,*** *Проректор по научной работе, профессор, д.т.н., Московский технический университет связи и информатики, Москва, Россия. orcid.org/0000-0003-3321-4497*

***Дымкова Светлана Сергеевна,*** *к.т.н., Московский технический университет связи и информатики, Москва, Россия. orcid.org/0000-0003-1945-9850*

***Фатхулин Тимур Джалилевич,*** *доцент, к.т.н., Московский технический университет связи и информатики, кафедра "Математическая кибернетика и информационные технологии", Москва, Россия. orcid.org/0000-0003-0998-1055*