

A Review of Convolutional Neural Network based Image Denoising Algorithms

Mengke Wang^{1,*}

¹ School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan 454000, PR China

Abstract

Currently, image-denoising algorithms based on convolutional neural networks (CNN) have been widely used and have achieved good results. Compared with traditional image-denoising methods, it has powerful learning ability and efficient algorithms. This paper summarizes traditional denoising methods and CNN-based image denoising methods, and introduces the basics of image denoising in detail, which is helpful for readers who are starting with image denoising processing. In addition, this paper also summarizes some commonly used datasets in the field of image processing, which makes it easier for us to denoise images. Finally, some suggestions for improving the performance of CNN image denoising are presented, and possible future research directions are discussed.

Keywords: Convolutional Neural Network, Denoising Algorithms, Image Processing, Deep Learning.

Received on 17 June 2023, accepted on 25 June 2023, published on 10 July 2023

Copyright © 2023 Wang *et al.*, licensed to EAI. This is an open access article distributed under the terms of the [CC BY-NC-SA 4.0](#), which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetel.v8i3.3461

1. Introduction

As technology reaches into all aspects of life, the number of digital images is increasing [1]. However, the quality of images varies, so the processing of digital images has become a problem [2]. Image processing techniques include image denoising, image enhancement, image restoration, and so on [3]. Image denoising is the basic problem of image processing and is essential for the subsequent processing of images, as a prerequisite for other processing operations. This is because only a clear image of high quality will enable the subsequent processing to achieve the desired result. A priori knowledge of image noise can help us to remove the noise better [4]. This section introduces the basics of noise, including the sources of noise, noise classification, and the application of CNN in noise removal.

1.1. Sources and Classification of Noise

When forming an image, we want the brightness of all parts of the image to be uniform, except for the parts that need to form the image [5]. However, reality is often different from the ideal state and some factors that are not required to form the image will also produce variations in brightness. This variation is random and usually results in a loss of image quality. We refer to this random variation as image noise [6]. For a picture, noise is the excess signal in what we are looking at. It is then visually represented as isolated pixel dots. In general, it is what reduces the observability of image features and makes the image unclear. Noise is generated mainly in the two processes of image signal acquisition and image signal transmission. The former is caused by differences in the sensor during acquisition, due to differences in the sensor material or circuit structure, etc. The latter is mainly due to noise caused by the unsatisfactory working of the transmission equipment [7].

*Corresponding author. Email: mengkewang@home.hpu.edu.cn

There are various ways of classifying image noise, for example, noise can be divided into two main categories [8], additive noise and multiplicative noise, based on the relationship between the image noise and the image signal. There are also other methods of classification. For example, noise can be classified according to the type of probability distribution of the noise. Several common types of noise are described below [9].

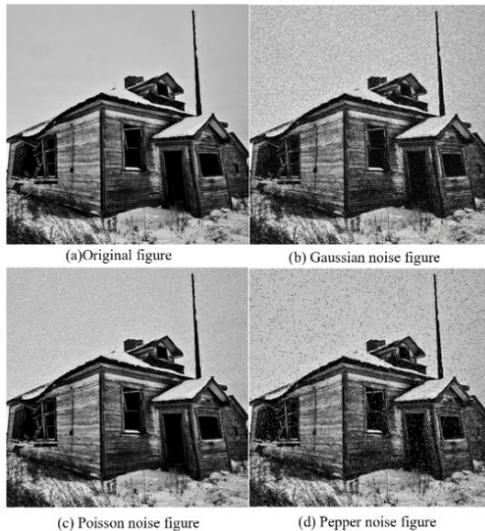


Figure 1 Original image and three typical noisy images

Gaussian noise: Gaussian noise is the most common type of noise [10]. Its probability density function obeys a Gaussian distribution, which is also known as a normal distribution. Its probability density function can be expressed as:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(z-\mu)^2}{2\sigma^2}} \quad (1)$$

Where z is the grey value of the image, the expectation and standard deviation of the grey value are divided into μ and σ . μ affects the position of the axis of symmetry and σ affects the width of the distribution of the Gaussian distribution curve.

Gaussian noise can be caused by several factors [11], such as a lack of brightness and uniformity of light in the environment in which the image is taken. Or the temperature of the sensor may be too high due to long periods of incessant operation. There is also a direct interaction between the various parts of the circuit that can lead to Gaussian noise. Using Figure 1 a as the original image, the effect of Gaussian noise is shown in Figure 1 b. **Poisson noise:** Poisson noise [12], also known as scattered particle noise, obeys the Poisson distribution. As shown in Figure 1 c. It arises due to the particle nature of light. It is well known that photons generated by a light source appear as visible dots on Complementary metal oxide semiconductors (CMOS) [13]. As shown in Figure 2.

Normally, the photons received by CMOS are proportional to the grey value of the pixel. However, not all emitted photons are received by the CMOS, and the frequency of emission is not always uniform. These can cause fluctuations in the grey value of the pixel, leading to the generation of scattered grain noise. Poisson noise is characterized by the fact that the light intensity is proportional to the noise [14].

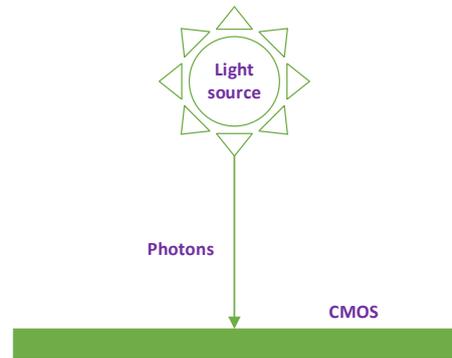


Figure 2 CMOS receiving photon principle

Pepper noise: Pepper noise, also known as impulse noise [15], is a discrete distribution of pixel dots, either black or white, in an image, caused mainly by electromagnetic interference that randomly alters some pixels of the image. As shown in Figure 1 d. The most effective method of removing pepper noise is currently the median filter [16]. In addition to this, noise can be divided into external noise and internal noise according to the cause of the noise. Noise can be divided into smooth and non-smooth noise according to different statistical characteristics.

1.2. Application of CNN in Image Denoising

With the rapid development of computer networks, computer performance has been improved to an unprecedented extent, laying the foundation for the development of deep learning [17]. Deep learning is being applied in an increasingly wide range of fields [18]. As a low-level task in the field of computer vision, image denoising has also been influenced by deep learning. Many excellent denoising algorithms have been generated that surpass traditional denoising algorithms in terms of denoising results [19].

The results of deep learning [20] rely on data, and a large amount of simulation data provides the possibility of applying deep learning to image denoising. This is especially true for images with complex backgrounds, where traditional algorithms often do not yield good results even after a lot of effort [21]. But deep learning can solve this problem very well by simply letting the network model learn from noisy samples to get a clean image [22-25]. By learning the mapping between noisy images and clean

images, a good denoising model can be trained to get good results.

2. Traditional Denoising Algorithms

2.1. Filter Category

Classical filtering and denoising algorithms for images can be broadly classified into two categories: spatial domain and frequency domain [26].

2.1.1. Spatial Domain

Spatial domain-based filtering is a direct operation on image pixels, processing each pixel in the image [27].

Domain operations are performed on pixels in image space with the help of Windows. The spatial domain can be subdivided into local and non-local filtering [28]. Typical local filters include:

Mean value filtering: Mean filtering is also known as the arithmetic mean filtering [29]. As shown in Figure 3, this reduces the sharpness of an image by using the domain average of a pixel instead of the pixel value at that point. But mean filtering also has very obvious disadvantages. For example, it is not very effective at removing noise, and it also blurs image detail and reduces image contrast.

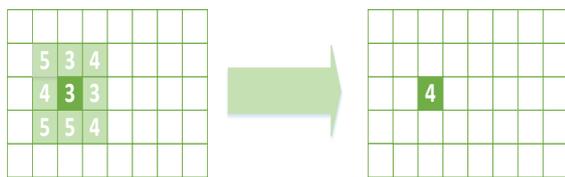


Figure 3 Mean filter calculation method

Gaussian filtering: Gaussian filtering is done by taking a particular pixel value as the center and then using a convolution kernel to apply a weighted average to the pixels in the field around it [30]. If x is used as the centre in the window, then the weight of a particular point y can be found by equation.

$$G(y) = \frac{1}{2\pi\sigma^2} e^{-\frac{\|x-y\|^2}{2\sigma^2}} \quad (2)$$

Where σ is the standard deviation of the pixel values in the window. As the value of σ increases, the filtering effect becomes better. But the less clear the image is. The Gaussian filtering weights are characterized by the fact that the further away from the center the weights are, the smaller they are [31]. But the weights are not randomly increased or decreased, its whole trend obeys the Gaussian function. Finally, the resulting pixel value is used as the new pixel value. Compared to mean filtering the result is better. However, Gaussian filtering smooths out the edge information for the sake of noise removal, thus causing blurring of the image.

Median filtering: Median filtering is a process where the pixel values within a certain window are selected and arranged in ascending order, and the median of these values is selected as the new central pixel value [16, 32]. The whole process can be represented in Figure 4, using a window size of 3×3 and a median of 4 instead of the original central pixel value of 6. Median filtering is particularly effective for pretzel noise, as the pixel values of pretzel noise are usually located at the ends of the aligned pixel values. As a result, it is more likely to be removed.

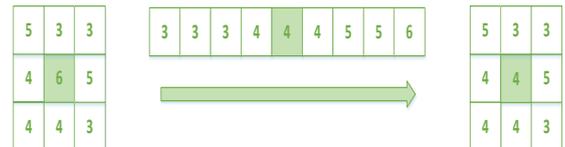


Figure 4 Median filter calculation method

Bilateral filtering: The best feature of the bipartite filter compared to the other filters described earlier is its ability to do edge preservation. It is a non-linear filtering method [33]. Not only the proximity of the spatial domain is taken into account, but also the similarity between pixel values. By weighing up the two, it compensates for the disadvantages of Gaussian filtering, allowing good denoising while preserving image edge detail.

Bootstrap filtering: The bootstrap filter is an edge-preserving filter, as is the bilateral filter [34]. Edge protection is possible because two adaptive adjustment factors are added to adjust the linear transformation between the input image and the guide image, and the adjustment factors can determine the gradient retention capability. Its basic idea is to use a bootstrap image to process the input image. The process can be expressed using the equation:

$$x_i = \sum_j W_{ij}(z) y_j \quad (3)$$

Where x is the output image, y is the input image, z is the guide image, i, j are the pixel index values and W is the filtering weight.

A typical representative of non-local filtering is the Non-Local Means (NLM) algorithm.

The main difference between the NLM algorithm and the previous filtering algorithms is that the NLM algorithm takes into account the pixel value information outside a specific window and makes full use of the original image information for image denoising by using the pixel distribution information of the whole image [35]. The basic idea is to chunk the image and process it in image blocks. The NLM algorithm is very effective in removing Gaussian noise.

2.1.2. Frequency Domain

Transform domain denoising starts by transforming the image from the spatial domain to the transform domain,

where the image is denoised [36]. This transformation process is carried out utilizing many integral transforms, such as the Fourier transform, the discrete cosine transform and the wavelet transform. This is achieved by separating the signal from the noise in the transform domain and then filtering out the noise. In this way, the purpose of denoising is achieved. The wavelet transform is the classical transform domain denoising.

The wavelet transform is a modification of the Fourier transform, in which the Fourier transform in a segment of the signal obtains which frequencies are contained and the type of frequency [37]. But if it is a non-smooth signal, we also need to know at which point in time the different frequencies occur. This is how we can know how the frequency of the signal changes with time. This is what we often call time-frequency analysis. The advantage of the improved wavelet transform over the Fourier transform is the multi-scale refinement of the signal to be processed and the localized analysis of the signal. The signal can be decomposed according to different sizes of frequencies. The basic idea of wavelet denoising is to separate the signal from the noise. This is because the statistical properties of the signal and the noise in the image are different after the wavelet transform. So a threshold can be set to separate noise with smaller wavelet coefficient amplitudes [38]. The steps of wavelet threshold denoising are shown in Figure 5:

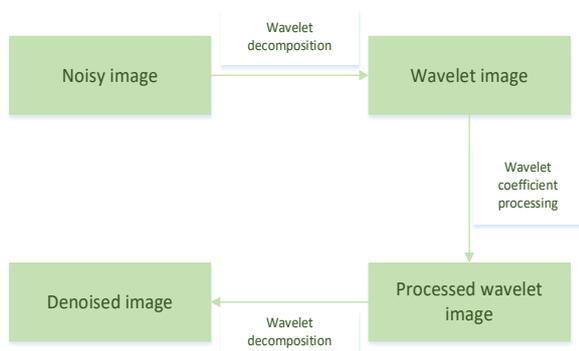


Figure 5 Principle of the wavelet denoising process

2.1.3. Hybrid Domain

The most typical denoising method for the hybrid domain is the Block-matching and 3D filtering (BM3D) algorithm [39]. The algorithm is the most effective among the traditional denoising algorithms and is the benchmark for denoising algorithms. The algorithm is divided into two main parts. The first part is to use a hard threshold in the collaborative filtering stage to find similar blocks in the whole image range, and then stack these different similar blocks into different blocks. Several similar blocks are formed into arrays for DCT transformation, and then through the filtering of hard threshold, the inverse transform is then put back into the image where it came from [40]. The second step is similar to the first step, except that the Wiener filtering process is applied to the image

group after the first step.

2.2. Sparse Expression Classes

Sparse representation is achieved by considering the image to be processed as a combination of a clear image and a noisy image [41]. Since the clear image is sparse and the noisy image is not sparse. Therefore, the image is reconstructed by extracting the sparse components of the image to be processed. In this process, the noise is discarded as a residual of the reconstructed image and the image to be processed. The aim is thus to achieve denoising. The most typical denoising algorithm for sparse representation is the Singular value decomposition K-SVD algorithm [42], which is a generalization of the k-means algorithm and is a dictionary learning algorithm for sparse representation. Simply put, sparse representation is the process of representing complex information with a small amount of simple information, and the implementation of this process requires a dictionary to complete. And the K-SVD algorithm is that a good dictionary can be designed.

2.3. Clustering Low Rank

Clustering low-rank algorithms have a wide range of uses in image processing [43]. For example, image segmentation, classification, and denoising. Its algorithm principle in denoising is similar to that of sparse representation methods. It treats the image to be processed as a matrix of similar samples, which is of low rank, but the noise is not of low rank [44]. Therefore, clustering low rank allows the denoising of the image to be processed by this feature. Weighted Kernel Norm Minimization (WNNM) [45] is one of the representative algorithms. This algorithm is good for image detail retention and is a conventional denoising algorithm with good results.

In conclusion, traditional denoising algorithms can remove image noise to a certain extent though. But there are also many disadvantages, for example, the tendency to lose details in the image and the appearance of blurred images. There is also the fact that the choice of parameters is not an easy matter of opinion. For some complex noise types, it is not possible to achieve good results.

3. Image Denoising Based on CNN

3.1. Introduction to CNN

We all know that CNN is a kind of neural network. CNN is a type of neural network that mimics the neural system of the human brain and is a form of machine learning. There are other types of connectivity, such as Generative Adversarial Networks (GAN) and recurrent neural networks (RNN) [46, 47]. They can act like the human brain and can make simple decisions about things. CNN

has been widely used in the field of image processing since its introduction and has achieved good results in solving many challenges in the field of image processing [48, 49]. The convolutional layer is mainly used to extract the feature information of the input image, through which the computer can identify the semantic information of the image. The pooling layer is used for feature dimensionality reduction, which compresses the number of parameters to improve the computational efficiency without corrupting the recognition results and improves the performance of the model [50]. The fully connected layer is mainly used for classification, combining the extracted local features into global features.

3.2. CNN Architecture for Denoising

LeNet: Following the great success of CNN in the field of image denoising, more and more CNN network architectures have been designed and refined with increasing efficiency. LeNet was the first network architecture proposed by [Cun, et al. \[51\]](#), it is a synthesis of a family of network architectures including LeNet1-5, with LeNet-5 being the most stable. If the input layer is not included, the network structure has a total of 7 network layers. The network structure is shown in Figure 6. It includes three convolutional layers, two down-sampling layers, one fully connected layer, and one output layer. It performs well in classifying handwritten digital datasets by using different numbers of convolutional kernels for feature extraction.

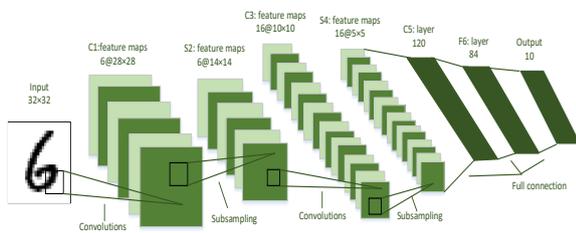


Figure 6 LeNet network structure

AlexNet: As computer performance improved, a deeper CNN model, AlexNet, was proposed by [Krizhevsky, et al. \[52\]](#). The model was first proposed for the classification of images as well and was a milestone-like existence. For the first time, stacked convolutional layers were used for feature extraction, and the exact steps in the model are shown in Figure 7. The network has 5 convolutional layers, a maximum pooling layer after 1, 3, and 5 layers of convolution, and 3 fully connected layers. The output of the network is the probability values of the image categories. AlexNet's innovations over the LeNet network include the use of the ReLU activation function, which solves the problem of gradient disappearance as the depth of the network increases.

Another innovation is the Local Response Normalization (LRN) operation added between the convolution and

pooling layers. By adding LRN the risk of overfitting the network can be reduced and the network's ability to adapt to new incoming samples can be improved. In addition, AlexNet also performs a series of transformations on sample images through data augmentation to increase the diversity of samples and improve the network's generalization ability. The inclusion of the Dropout technique is also one of the highlights of the network. In addition, the model uses overlapping maximum pooling in the CNN, achieving better results than average pooling, resulting in a richer set of extracted features. In conclusion, the introduction of AlexNet has greatly contributed to the development of the image processing field.

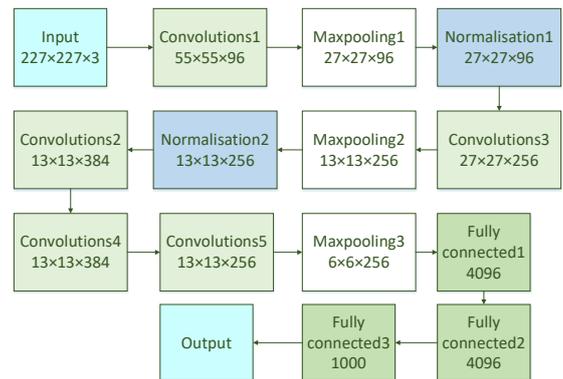


Figure 7 AlexNet network structure

VGGNet: Following AlexNet, [Simonyan and Zisserman \[53\]](#) proposed a network model that won an award at the 2014 ILSVRC. The most important feature of VGGNet is that it uses several successively smaller convolutional kernels instead of large convolutional kernels, increasing the depth of the network without losing perceptual fields, making the network more capable of learning. It also reduces the number of parameters and reduces the network overhead. The structure of the VGG-16 model is shown in Figure 8. It contains 13 convolutional layers and 3 fully connected layers, and the entire network structure uses a 3×3 convolutional kernel and a 2×2 pooling layer. However, VGGNet uses a large number of parameters, so VGGNet is relatively more resource-intensive.

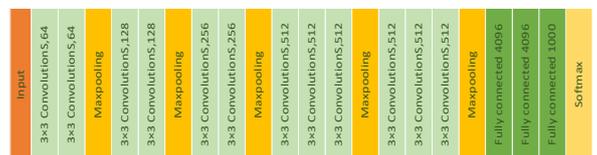


Figure 8 VGG network structure

GoogLeNet: To further improve the performance of the network, GoogLeNet is proposed. [Szegedy, et al. \[54\]](#) proposed the Inception method to study the width of the network in terms. The Inception module is shown in Figure 9. The Inception method is to assemble several convolutions or pooling into one module and use these

modules to assemble the model structure. By combining several convolutions or pooling it is possible to extract multi-scale feature information of the image, allowing the network to select different-size convolution kernels according to different images. The flexibility and efficiency of the network is improved.

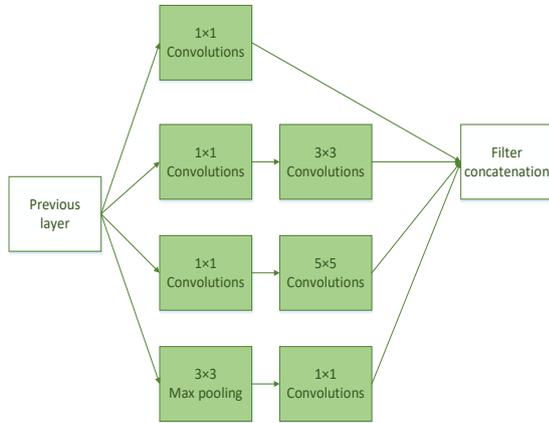


Figure 9 Inception module

ResNet: However, as the depth and width of the network increase, the network tends to degenerate in the form of disappearing gradients, exploding gradients, and overfitting. ResNet was proposed by He, et al. [55] to solve these problems. The proposed model allows the network to reach a new level of depth, mainly thanks to the inclusion of residual blocks. The structure of the residual block is shown in Figure 10. The so-called residual is the difference between the observed and estimated values. With the residual mapping, a better result can be chosen between the shallow model and the deep model, and degradation of the model is avoided.

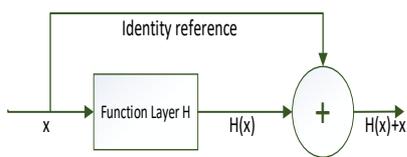


Figure 10 Structure of the residual block

3.3 Typical CNN Denoising Model

DnCNN: Denoising Convolutional Neural Network (DnCNN) is the simplest network model proposed by Zhang, et al. [56] for image denoising. It belongs to the feed-forward network model. It is a simple model with low memory overhead but effective. The paper uses additive noise to train the model, assuming that the rational noise-free image is C , the image after adding noise is N , and the noise is represented by V . The relationship between noise and image is then

$$N = C + V \tag{4}$$

The task of the denoising model then translates into separating C from N .

As shown in Figure 11, DnCNN uses a deeper neural net compared to its previous denoising method. ReLU is used as the activation function. To speed up model convergence, further, improve model stability, and alleviate the problem of gradient dispersion, Chinese also incorporates batch normalization. The model separates the image from the noise through a hidden layer to obtain a residual image, and then a clean image through the residual image.

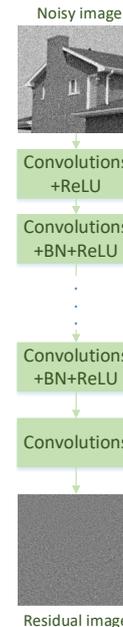


Figure 11 DnCNN model denoising process

FFDNet: The FFDNet proposed by Zhang, et al. [57] is also one of the very classic papers in the field of image denoising. The success of CNN in the field of image processing is mainly due to its powerful modeling capabilities, as well as its ability in network training and optimization. However, most denoising models tend to use well-defined noise levels, which results in models that perform well at specific noise levels and poorly at other noise levels. The model solves this problem. The most significant feature of the model is the inclusion of noise level mapping. The parameters of the model change as the noise level is transformed. When the noise level is too high, the denoised image tends to return with the image details smoothed out, and the image quality is improved by using an orthogonal initialization method on the convolutional filter. In contrast to DnCNN, this model uses depth-to-space and space-to-depth to up-sampling once on top of DnCNN. If the input image size is $H \times W \times C$, after down-sampling, it becomes four images of $\frac{H}{2} \times \frac{W}{2} \times 4C$ large. Where W and H are the width and height of the image and C is the number of image channels. This method reduces the network parameters, resulting in an increased perceptual field and better model performance.

CBDNet: The CBDNet model proposed by [Guo, et al. \[58\]](#) differs from the above two denoising models in that it can effectively denoise real images. The model consists of two main sub-networks, noise estimation, and non-blind denoising so that blind denoising of images can be performed without knowing the noise level. In addition, the model combines synthetic noisy images and real noisy images for training, rather than using a single prescribed noisy image of either kind. This improves the performance of the model. The asymmetric loss function is also a highlight of this paper, providing interactive experimental results that improve the robustness of the model. In conclusion, this model provides an idea for solving the problem of complex realistic noisy images, which is worthwhile to learn from.

GAN: In recent years, the development of GAN in the field of image processing is also not negligible [\[59\]](#). Usually, in image denoising, the samples trained by GAN are noisy images. However, this method does not guarantee any loss of image content. This is because the model learns the feature information in the original image by learning it and retaining it as much as possible. It still does not avoid loss during denoising. The network model GAN-CNN Based Blind Denoiser (GCBD) improves this problem by generating similar noise. The model is also divided into two parts, as shown in Figure 12. Firstly, the GAN is trained to estimate the noise distribution on the input image and generate noise samples. The second part uses the samples from the first part to construct a paired dataset and then uses the CNN to denoise the given image.

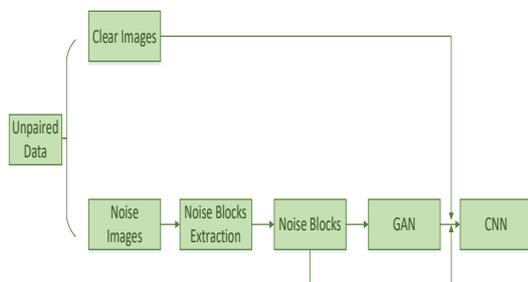


Figure 12 GCBD model denoising process

NBNet: [Cheng, et al. \[60\]](#) proposed a new denoising method for image adaptive projection. In layman's terms, the method splits the noisy image into a set of bases in space, then finds the set of bases where the noise in the image is located and removes that part, thus completing the task of denoising the image.

4. Dataset and Result Analysis Evaluation

4.1 Typical Denoising Dataset

In the field of image processing, there are three main ways

in which datasets are built [\[61\]](#). Using a high-quality image from an existing image database is adjusted and then the noise is added manually. The second way is to use two images with the same brightness and different International Organization for Standardization(ISO). The high ISO is the noisy image and the low ISO is the ground truth. the third way is to take multiple shots of the same scene and then weigh and merge the filtered images into the ground truth. the dataset is further divided into a training dataset and a test dataset. Table 1 and Table 2 summarize several commonly used training and test datasets.

(1) Training dataset

Datasets	Image Type	Number	Resolution size
PolyU [62]	RGB	100 sheets	512×512
SSID [63]	RGB	200 sheets	Different resolutions for different filming devices
DND [64]	RGB	40 sheets	512×512
RENOIR [61]	RGB	120 sheets	Different resolutions for different filming devices

Among the things that PolyU is a real image dataset containing real image noise from different scenes, taken by different cameras in different scene settings. SSID was taken in different lighting conditions with different smartphone models. DND including 50 indoor/outdoor scenes shot. RENOIR includes 120 indoor and outdoor dark scenes, two noise, and two low-noise images per scene.

(2) Test dataset

Datasets	Image Type	Number	Resolution size
Set12 [56]	Grayscale images	12 sheets	256×256 or 512×512
CSet8	Color images	8 sheets	256×256 or 512×512
Kodak24	Color images	24 sheets	500×500
McMaster [65]	Color images	18 sheets	500×500
RNI6 [57]	Grayscale	6	500×500

	images	sheets	
RN15	Color images	15 sheets	500×500 or 481×321
BSD68[24]	Grayscale images	68 sheets	321×481 or 481×321
CBSD68	Color images	68 sheets	321×481 or 481×321

Among the things that Set12 is the most commonly used dataset in digital image processing and contains grey images of different scenes. CSet12 is a color image of Set12. Kodak24 includes images of buildings, people, animals, the seaside, etc. McMaster includes fruit, flowers, sofas, etc. RNI6 includes portraits, buildings, etc. RN15 includes portraits, flowers, animals, mugs, etc. BSD68 includes buildings, portraits, scenes, animals, plants, etc. CBSD68 is a color image of BSD68.

4.2 Evaluation Index of Denoising Effect

After an image has been denoised, image quality evaluation criteria are usually introduced to evaluate the goodness of the denoising model. There are two main categories of evaluation criteria: subjective criteria and objective criteria. This paper focuses on several objective evaluation criteria [66].

1. MSE

Mean Square Error (MSE) evaluates the quality of an image by calculating the square of the difference in pixels between the model's predicted value and the true value, then summing the squares of the differences between all the pixels in the two images and averaging them [67]. It is expressed using the formula as follows:

$$MSE = \frac{1}{P \times R} \sum_{x=0}^{P-1} \sum_{y=0}^{R-1} [m(x, y) - n(x, y)]^2 \quad (5)$$

P and R are the total number of pixel values of the two images respectively, $m(x, y)$ is the true clean image, and $n(x, y)$ is the denoised image. a smaller value of MSE means a smaller difference between the predicted and true values, which means better denoising and better performance of the model.

2. PSNR

The Peak Signal Noise Ratio (PSNR) is the most important metric for evaluating image quality, not only for image denoising but also for image super-resolution reconstruction [68]. PSNR represents the ratio of the maximum possible power of the signal to the power of the noise, the larger the ratio the greater the signal share, and the smaller the noise share, the better the quality of the image. It is measured in decibels (dB), and generally speaking, if the PSNR exceeds 40dB, the image quality is

very good. If MAX_i indicates the maximum grey scale value and MSE indicates the mean square error, then the PSNR is expressed in the following formula:

$$PSNR = 10 \log_{10} \left(\frac{MAX_i^2}{MSE} \right) \quad (6)$$

3. SSIM

Structural similarities (SSIM) measures the similarity of two images and is a commonly used metric for image quality evaluation [69]. Assuming input images X and Y , the SSIM for images X and Y is calculated as

$$SSIM(X, Y) = \frac{(2\mu_X\mu_Y + a_1)(2\sigma_{XY} + a_2)}{(\mu_X^2 + \mu_Y^2 + a_3)(\sigma_X^2 + \sigma_Y^2 + a_4)} \quad (7)$$

μ_X and μ_Y in the formula are the standard deviations of X and Y , σ_X^2 and σ_Y^2 are the variances of X and Y , σ_{XY} is the covariance of X and Y , and $a_1 \square a_2 \square a_3$ and a_4 are constants. In general, the SSIM values are in the range $[-1, 1]$, within which the larger the SSIM value, the better the model is at denoising the image.

4.3 Methods for Improving the Performance of CNN Image Denoising

To obtain better denoising results, we can also optimize in the following directions.

1. Data enhancement

Data augmentation is the process of making data more valuable without changing the substance of the data. The image data is pre-processed and augmented before it is fed into the model. Typically, we perform image transformations such as rotation, translation, scaling, and cropping. We can also change the color of the image or add noise to the image. This approach allows the model to learn more features from a complex image, thus improving model performance.

2. Loss function

It is always known that the choice of a suitable loss function is very important for the model. We can try to use some hybrid loss functions. There are other loss functions such as adversarial loss, perceptual loss, and so on. The loss function can be used to improve the denoising effect and optimize the performance of the model.

3. Adding new techniques

Adding some new techniques to the CNN model can optimize the model performance, such as self-attentive mechanism, residual module, and so on. These have already yielded good results in image denoising, and we need to explore more new techniques to help the development of image processing.

Design of better network structures: Improving the performance of the network model requires deeper, wider, and better architectures for the network structure. Denoising can also be improved by adding more network branches. Aspects of image processing tasks are inherently interconnected, and network architectures proposed for

other tasks can also be borrowed and fused with image denoising to help the development of the image denoising field.

5. Conclusion

This paper provides a detailed introduction to the basics related to the field of image denoising and summarizes the development of image denoising, from traditional denoising algorithms to neural network-based algorithms [70]. However, this paper is a theoretical summary of the denoising algorithm and lacks the support of experimental data, which will be investigated later. Over the past few decades, the field of image denoising has achieved great success and the level of denoising has continued to improve. But our quest does not stop there, and we should continue to explore better algorithms for denoising. There are still many problems to be solved and explored in the future of image denoising. For example, in some complex scenes, the currently used denoising methods are not able to fully exploit their usefulness and cannot handle detailed areas cleanly. For some mixed types of noise, no algorithm can completely solve the problem, and only specific predicted noise can be denoised. In addition, deep learning-based algorithms require high-performance hardware to compute large data sets, which is a long time and costly affair. These issues are the general direction of future developments in image processing.

References

- [1] M. S. El_Tokhy, "Development of precise forgery detection algorithms in digital radiography images using convolution neural network," *Applied Soft Computing*, vol. 138, 2023.
- [2] S. Bhattacharyya, "A Brief Survey of Color Image Preprocessing and Segmentation Techniques," *Journal of Pattern Recognition Research*, vol. 6, no. 1, pp. 120-129, 2011.
- [3] N. A. Vinnichenko, A. V. Pushtaev, Y. Y. Plaksina, and A. V. Uvarov, "Performance of Background Oriented Schlieren with different background patterns and image processing techniques," *Experimental Thermal and Fluid Science*, vol. 147, 2023.
- [4] Y. Zhao, H. M. Guo, and S. A. Billings, "Application of Totalistic Cellular Automata for Noise Filtering in Image Processing," *Journal of Cellular Automata*, vol. 7, no. 3, pp. 207-221, 2012.
- [5] J. Anaya and A. Barbu, "RENOIR - A dataset for real low-light image noise reduction," *Journal of Visual Communication and Image Representation*, vol. 51, pp. 144-154, Feb 2018.
- [6] G. Cui, H. Feng, Z. Xu, Q. Li, and Y. Chen, "No-reference image noise estimation based on noise level accumulation," *Optical Review*, vol. 23, no. 2, pp. 208-219, 2016.
- [7] Z. X. Zhong, Z. S. Jiang, Y. H. Long, and X. Zhan, "Analysis on the Noise for the Different Gearboxes of the Heavy Truck," *Shock and Vibration*, vol. 2015, 2015, Art no. 476460.
- [8] F. Liu, Q. Z. Song, and G. H. Jin, "The classification and denoising of image noise based on deep neural networks," *Applied Intelligence*, vol. 50, no. 7, pp. 2194-2207, Jul 2020.
- [9] F. A. Wichmann, D. I. Braun, and K. R. Gegenfurtner, "Phase noise and the classification of natural images," *Vision Res*, vol. 46, no. 8-9, pp. 1520-9, Apr 2006.
- [10] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data," *IEEE Trans Image Process*, vol. 17, no. 10, pp. 1737-54, Oct 2008.
- [11] J. Song, "SPDEs with Colored Gaussian Noise: A Survey," *Communications in Mathematics and Statistics*, vol. 6, no. 4, pp. 481-492, 2018.
- [12] X.-d. Wang, X.-c. Feng, W.-w. Wang, and W.-j. Zhang, "Iterative reweighted total generalized variation based Poisson noise removal model," *Applied Mathematics and Computation*, vol. 223, pp. 264-277, 2013.
- [13] P. K. Nayak, J. A. Caraveo-Frescas, Z. Wang, M. N. Hedhili, Q. X. Wang, and H. N. Alshareef, "Thin film complementary metal oxide semiconductor (CMOS) device using a single-step deposition of the channel layer," *Sci Rep*, vol. 4, p. 4672, Apr 14 2014.
- [14] J. Salmon, Z. Harmany, C.-A. Deledalle, and R. Willett, "Poisson Noise Reduction with Non-local PCA," *Journal of Mathematical Imaging and Vision*, vol. 48, no. 2, pp. 279-294, 2013.
- [15] F. Dong, Y. Chen, D.-X. Kong, and B. Yang, "Salt and pepper noise removal based on an approximation of l0 norm," *Computers & Mathematics with Applications*, vol. 70, no. 5, pp. 789-804, 2015.
- [16] M. Welk, "Multivariate Median Filters and Partial Differential Equations," *Journal of Mathematical Imaging and Vision*, vol. 56, no. 2, pp. 320-351, Oct 2016.
- [17] D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick, "Neuroscience-Inspired Artificial Intelligence," *Neuron*, vol. 95, no. 2, pp. 245-258, Jul 19 2017.
- [18] Y. Zhang, S. C. Satapathy, L. Y. Zhu, J. M. Gorriz, and S. Wang, "A Seven-Layer Convolutional Neural Network for Chest CT-Based COVID-19 Diagnosis Using Stochastic Pooling," *IEEE Sens J*, vol. 22, no. 18, pp. 17573-17582, Sep 2022.
- [19] Y. Kim, J. W. Soh, G. Y. Park, and N. I. Cho, "Transfer Learning From Synthetic to Real-Noise Denoising With Adaptive Instance Normalization," presented at the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
- [20] S.-H. Wang, "SOSPCNN: Structurally Optimized Stochastic Pooling Convolutional Neural Network for Tetralogy of Fallot Recognition," *Wireless Communications and Mobile Computing*, vol. 2021, p. 5792975, 2021/07/02 2021, Art no. 5792975.
- [21] Y. Yan, "A survey of computer-aided tumor diagnosis based on convolutional neural network," *Biology*, vol. 10, no. 11, 2021, Art no. 1084.
- [22] D. F. Wu, K. Kim, and Q. Z. Li, "Low-dose CT reconstruction with Noise2Noise network and testing-time fine-tuning," *Medical Physics*, vol. 48, no. 12, pp. 7657-7672, Dec 2021.
- [23] H. Chen *et al.*, "Low-Dose CT With a Residual Encoder-Decoder Convolutional Neural Network," *IEEE Trans Med Imaging*, vol. 36, no. 12, pp. 2524-2535, Dec 2017.
- [24] R. J. Ma, S. Y. Li, B. Zhang, and H. F. Hu, "Meta PID

- Attention Network for Flexible and Efficient Real-World Noisy Image Denoising," *Ieee Transactions on Image Processing*, vol. 31, pp. 2053-2066, 2022.
- [25] A. E. Ilesanmi and T. O. Ilesanmi, "Methods for image denoising using convolutional neural network: a review," *Complex & Intelligent Systems*, vol. 7, no. 5, pp. 2179-2198, Oct 2021.
- [26] X. Sun, Z. Zhu, X. Liu, Y. Shang, and Q. Yu, "Frequency-spatial domain based salient region detection," *Optik*, vol. 126, no. 9-10, pp. 942-949, 2015.
- [27] S. Wang, "Advances in data preprocessing for biomedical data fusion: an overview of the methods, challenges, and prospects," *Information Fusion*, vol. 76, pp. 376-421, 2021.
- [28] P. Li, X. Liu, and H. Xiao, "Quantum image median filtering in the spatial domain," *Quantum Information Processing*, vol. 17, no. 3, 2018.
- [29] P. Nair and K. N. Chaudhury, "Fast High-Dimensional Bilateral and Nonlocal Means Filtering," *IEEE Trans Image Process*, Oct 31 2018.
- [30] M. L. Psiaki, "The Blind Tricyclist Problem and a Comparative Study of Nonlinear Filters A CHALLENGING BENCHMARK FOR EVALUATING NONLINEAR ESTIMATION METHODS," *Ieee Control Systems Magazine*, vol. 33, no. 3, pp. 40-54, Jun 2013.
- [31] J. Ala-Luhtala, S. Särkkä, and R. Piché, "Gaussian filtering and variational approximations for Bayesian smoothing in continuous-discrete stochastic dynamic systems," *Signal Processing*, vol. 111, pp. 124-136, 2015.
- [32] S. A. Villar, S. Torcida, and G. G. Acosta, "Median Filtering: A New Insight," *Journal of Mathematical Imaging and Vision*, vol. 58, no. 1, pp. 130-146, 2016.
- [33] S. Morillas, V. Gregori, and A. Sapena, "Fuzzy bilateral filtering for color images," in *Image Analysis and Recognition, Pt 1*, vol. 4141, A. Campilho and M. Kamel, Eds. (Lecture Notes in Computer Science, 2006, pp. 138-145.
- [34] N. Zheng, Y. Pan, X. Yan, and R. Huan, "Local weight mean comparison scheme and architecture for high-speed particle filters," *Electronics Letters*, vol. 47, no. 2, pp. 142-+, Jan 2011.
- [35] M. Liu, Z. Y. Wang, and S. W. Ji, "Non-Local Graph Neural Networks," *Ieee Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 10270-10276, Dec 2022.
- [36] M. S. Sonia and S. Sumathi, "A Comparative Analysis of Wavelet Transforms for Denoising MRI Brain Images," presented at the 2022 International Conference on Power, Energy, Control and Transmission Systems (ICEPTS), 2022.
- [37] I. Y. Orea-Flores, F. J. Gallegos-Funes, and A. Arellano-Reynoso, "Local Complexity Estimation Based Filtering Method in Wavelet Domain for Magnetic Resonance Imaging Denoising," *Entropy*, vol. 21, no. 4, Apr 2019, Art no. 401.
- [38] B.-q. Chen, J.-g. Cui, Q. Xu, T. Shu, and H.-l. Liu, "Coupling denoising algorithm based on discrete wavelet transform and modified median filter for medical image," *Journal of Central South University*, vol. 26, no. 1, pp. 120-131, 2019.
- [39] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans Image Process*, vol. 16, no. 8, pp. 2080-95, Aug 2007.
- [40] B.-J. Zou, Y.-D. Guo, Q. He, P.-B. Ouyang, K. Liu, and Z.-L. Chen, "3D Filtering by Block Matching and Convolutional Neural Network for Image Denoising," *Journal of Computer Science and Technology*, vol. 33, no. 4, pp. 838-848, 2018.
- [41] J. Xiao, R. Zhao, and K.-M. Lam, "Bayesian sparse hierarchical model for image denoising," *Signal Processing: Image Communication*, vol. 96, 2021.
- [42] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *Ieee Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311-4322, Nov 2006.
- [43] M. Nejati, S. Samavi, H. Derksen, and K. Najarian, "Denoising by low-rank and sparse representations," *Journal of Visual Communication and Image Representation*, vol. 36, pp. 28-39, 2016.
- [44] Y.-D. Zhang, "Advances in Multimodality Data Fusion in Neuroimaging," *Information Fusion*, vol. 76, pp. 87-88, 2021.
- [45] Y. Xie, S. H. Gu, Y. Liu, W. M. Zuo, W. S. Zhang, and L. Zhang, "Weighted Schatten p-Norm Minimization for Image Denoising and Background Subtraction," *Ieee Transactions on Image Processing*, vol. 25, no. 10, pp. 4842-4857, Oct 2016.
- [46] I. W. Feng, W. P. Zhao, J. Li, J. Y. Lin, H. X. Jiang, and J. Zavada, "Correlation between the optical loss and crystalline quality in erbium-doped GaN optical waveguides," *Applied Optics*, vol. 52, no. 22, pp. 5426-5429, Aug 2013.
- [47] J. Y. Kong, Y. R. Gao, Y. J. Zhang, H. M. Lei, Y. Wang, and H. S. Zhang, "Improved Attention Mechanism and Residual Network for Remote Sensing Image Scene Classification," *Ieee Access*, vol. 9, pp. 134800-134808, 2021.
- [48] Y.-D. Zhang, "Improving ductal carcinoma in situ classification by convolutional neural network with exponential linear unit and rank-based weighted pooling," *Complex Intell. Syst.*, vol. 7, pp. 1295-1310, 2020/11/22 2021.
- [49] D. S. Guttery, "Improved Breast Cancer Classification Through Combining Graph Convolutional Network and Convolutional Neural Network," *Information Processing and Management*, vol. 58, 2, 2021, Art no. 102439.
- [50] X. Wang, H. Xuan, B. Evers, S. Shrestha, R. Pless, and J. Poland, "High-throughput phenotyping with deep learning gives insight into the genetic architecture of flowering time in wheat," *Gigascience*, vol. 8, no. 11, Nov 2019, Art no. giz120.
- [51] Y. L. Cun *et al.*, "Handwritten digit recognition with a back-propagation network," presented at the Proceedings of the 2nd International Conference on Neural Information Processing Systems, 1989.
- [52] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84-90, 2017.
- [53] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," p. arXiv:1409.1556 Accessed on: September 01, 2014. doi: 10.48550/arXiv.1409.1556 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2014arXiv1409.1556S>
- [54] C. Szegedy *et al.*, "Going Deeper with Convolutions," p. arXiv:1409.4842 Accessed on: September 01, 2014.

- doi: 10.48550/arXiv.1409.4842 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2014arXiv1409.4842S>
- [55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," p. arXiv:1512.03385 Accessed on: December 01, 2015. doi: 10.48550/arXiv.1512.03385 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2015arXiv151203385SH>
- [56] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising," *IEEE Transactions on Image Processing*, vol. 26, pp. 3142-3155, July 01, 2017 2017.
- [57] K. Zhang, W. Zuo, and L. Zhang, "FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising," *IEEE Transactions on Image Processing*, vol. 27, pp. 4608-4622, September 01, 2018 2018.
- [58] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward Convolutional Blind Denoising of Real Photographs," p. arXiv:1807.04686 Accessed on: July 01, 2018. doi: 10.48550/arXiv.1807.04686 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2018arXiv1807046866G>
- [59] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," p. arXiv:1411.1784 Accessed on: November 01, 2014. doi: 10.48550/arXiv.1411.1784 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2014arXiv1411.1784M>
- [60] S. Cheng, Y. Wang, H. Huang, D. Liu, H. Fan, and S. Liu, "NBNet: Noise Basis Learning for Image Denoising with Subspace Projection," p. arXiv:2012.15028 Accessed on: December 01, 2020. doi: 10.48550/arXiv.2012.15028 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2020arXiv201215028C>
- [61] J. Anaya and A. Barbu, "RENOIR - A Dataset for Real Low-Light Image Noise Reduction," p. arXiv:1409.8230 Accessed on: September 01, 2014. doi: 10.48550/arXiv.1409.8230 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2014arXiv1409.8230A>
- [62] J. Xu, H. Li, Z. Liang, D. C. Zhang, and L. Zhang, "Real-world Noisy Image Denoising: A New Benchmark," *ArXiv*, vol. abs/1804.02603, 2018.
- [63] A. Abdelhamed, S. Lin, and M. S. Brown, "A High-Quality Denoising Dataset for Smartphone Cameras," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1692-1700.
- [64] T. Plötz and S. Roth, "Benchmarking Denoising Algorithms with Real Photographs," p. arXiv:1707.01313 Accessed on: July 01, 2017. doi: 10.48550/arXiv.1707.01313 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2017arXiv170701313P>
- [65] L. Zhang, X. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *Journal of Electronic Imaging*, vol. 20, pp. 023016-023016-16, April 01, 2011 2011.
- [66] W. Shi, C. Zhu, Y. Tian, and J. Nichol, "Wavelet-based image fusion and quality assessment," *International Journal of Applied Earth Observation and Geoinformation*, vol. 6, no. 3-4, pp. 241-251, 2005.
- [67] H. Ren, M. El-Khamy, and J. Lee, "DN-ResNet: Efficient Deep Residual Network for Image Denoising," p. arXiv:1810.06766 Accessed on: October 01, 2018. doi: 10.48550/arXiv.1810.06766 [Online]. Available: <https://ui.adsabs.harvard.edu/abs/2018arXiv1810067666R>
- [68] W. Jifara, F. Jiang, S. Rho, M. Cheng, and S. Liu, "Medical image denoising using convolutional neural network: a residual learning approach," *The Journal of Supercomputing*, vol. 75, no. 2, pp. 704-718, 2017.
- [69] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, "Image Quality Assessment: Unifying Structure and Texture Similarity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2567-2581, 2022.
- [70] Y. Zhang, "Deep learning in food category recognition," *Information Fusion*, vol. 98, p. 101859, 2023/10/01/ 2023.