Development and Training of a Neural Network Filter for Satellite Images Processing

Nikita Andriyanov[1], Alexandr Kim[1]

[1] Financial University under the Government of the Russian Federation

Keywords: Image Processing, Image Filtering, Convolutional Networks, AutoEncoder, Satellite Images, Image Augmentation

Abstract

This paper is devoted to the study of the efficiency of using neural networks for filtering satellite images. The authors propose the use of convolutional noise suppressing autoencoders in order to minimize the filtering error variance. As part of the study, the architecture of the autoencoder was developed, optimal hyperparameters were selected and the resulting neural network model was trained. In addition, the paper compares the effectiveness of the proposed approach with traditional filtering algorithms such as Kalman filter and Wiener filter. Our models provide filtering efficiency gains of 3-4% at low noise levels (Signal-Noise-Ratio, SNR is 4 or more). The authors also investigated the effect of using data augmentations on improving the filtering quality. Experimental results showed that neural network models are able to outperform classical filters in terms of accuracy in processing real satellite images. Additionally, the paper studied the dependence of the filtering error variance on the number of training epochs of the neural network. The obtained results demonstrate that the developed neural network filter can be effectively applied for noise suppression on satellite images.

1. Introduction

The state-of-the-art capabilities of neural networks ensure their increasing application in image processing (Andriyanov et al., 2022 and Dosovitskiy et al., 2021) and, especially, in satellite image processing (Zhu et al., 2022). A number of studies show that the robustness of convolutional neural networks is under great question even when simple attacks are used (Andriyanov et al., 2021 and Lebedev et al., 2018). Therefore, one of the most important steps in data preprocessing is image filtering (Rubis et al., 2016). Indeed, removing noise in the preliminary stage leads to an improvement in the quality of the subsequent image processing, regardless of the problem to be solved. It is shown in (Vasiliev et al., 2017) that pre-filtering improves the efficiency of signal detection in images. Various regression filters are the most widely used. For example, the Kalman filter (Kalita and Lyakhov, 2022). Working with the frequency representation of the processed signal is given by Wiener filter (Chen et al., 2023).

The Kalman filter is an optimal linear recursive algorithm for estimating the state of dynamic systems in the presence of measurement noise and process noise. From a mathematical point of view, the Kalman filter is an algorithm based on the theory of optimal filtering, which allows to obtain optimal estimates of the internal states of a dynamic system described by a linear state equation in discrete time, in the presence of Gaussian white noise in measurements and process. The main property of the Kalman filter is its recursiveness. The main advantage of the Kalman filter is its ability to process noisy data, suppressing the noise and restoring the true state of the image.

Wiener filter or Wiener-Hopf filter is an optimal linear filter for processing stationary random signals based on the theory of optimal filtering. For regression-formed signals, the results of the Wiener filter cannot be worse than those of the Kalman filter. However, stationary and homogeneous images are rarely found in practice. Mathematically, the Wiener filter is a convolution of the input signal with the impulse response of the filter, which is determined on the basis of the correlation characteristics of the input and desired signals. In this case, the optimum is achieved in the sense of the minimum filtering variance error. It should be noted that the Wiener filter has more stringent requirements to the statistical properties of signals compared to the Kalman filter, which can work with non-stationary signals. In particular, its doubly stochastic modification was developed for the Kalman filter (Andriyanov et al., 2019).

At the same time, in real images, many different factors come into play, and the requirements that are necessary for successful filtering are often not met. Modifications of adaptive filters (Kleefeld et al., 2015), Gaussian (Wang et al., 2014) and median filters (Mursal et al., 2020) are used for this purpose. But they also cannot provide good results on heterogeneous data with high probability. Doubly stochastic filters (Dementyiev et al., 2020) occupy a special place in filtering images with complex structure. They provide parameter estimation with different filtering coefficients at different points of the image, which makes such filters more flexible.

Thus, image filtering is one of the most important tasks in image processing and computer vision, which is necessary for different applied tasks: from improving image quality to creating special effects on an image. At the same time, the capabilities of neural network-based image filtering are currently poorly understood. In this paper we propose to investigate the filtering efficiency of the autoencoder and compare the results with known models.

2. Data and Experiments

A dataset with satellite images was chosen for the study because the analysis showed that filtering is quite often applied to such data. The Kaggle dataset (Arjun Tyagi, 2020), was used. The image database contains satellite images belonging to several classes, including those from the surface of Mars and the Moon. The data distribution is presented in Table 1.

Because it will be necessary to obtain restored data it is interesting to compare different models on relative mean square error (MSE) and Structural Similarity Index (SSIM) metrics.

Publisher's note: Copernicus Publications has not received any payments from Russian or Belarusian institutions for this paper.

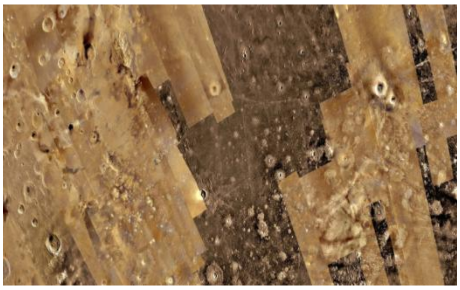
This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W5-2024-9-2024

Class	Number of images
Beach	100
Ice	73
Mars	100
Moon	100
Mountain	100
Ocean	100
River	100

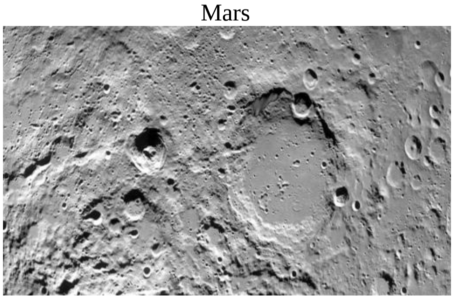
Table 1. Class distribution in the dataset

Figure 1 shows examples of images of the beach and Mars, moon and ocean respectively.

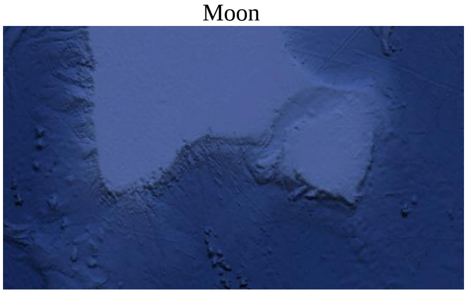
Beach



Mars



Moon



Ocean

Figure 1. Examples of images in dataset.

From the presented figures, it can be seen that the classes are quite different from each other. Usually the presented dataset is used for the classification task. However, we are going to solve the noise suppression task. Since it can be seen that the original images are quite clean, since the data collection is done in Google Earth Pro imaging conditions, we need to develop a noise addition model. In this case, we will evaluate errors separately within classes.

To obtain the noisy observations, we write down the following expression:

$$Z(i, j, c) = I(i, j, c) + N(i, j, c), \tag{1}$$

where Z is observed color image; I is original image before noise addition (reference image); N is white Gaussian noise with zero mean and arbitrary variance ; (i, j, c) is the set of functional parameters: pixel coordinate by row, pixel coordinate by column, color channel in accordance with RGB color scheme.

Thus, using expression (1), different models of noisy images can be obtained. At the same time it is possible to control the noise level in order to compare the quality of the developed filters in different conditions.

An autoencoder model (Bartlett et al., 2023) is proposed to solve the noise suppression problem. The autoencoder is trained by minimizing the error between the input and output images. In fact, for each image, it is necessary to match a pair of images: with and without noise. This allows the autoencoder to learn to extract meaningful features and suppress noise. For the comparison to be adequate, then several networks of the encoder must be trained for different levels of noise variance. After all, in Kalman filtering, it is assumed that the characteristics of the communication channel are known.

The process of noise suppression using an autoencoder can be visualized as follows:

Step 1. Encoding.
During this stage the first part of the autoencoder (directly encoder) works. Usually when working with images it is better to use convolutional layers that take into account spatial structure of images. At the input of the autoencoder there is an image with noise. The encoder part compresses this image into a compact low-dimensional representation in the form of a latent vector.

Step 2. Noise reduction.
It is achieved during training by selecting the weights of the encoder and decoder in such a way that the generated latent space allows the output image to be as close to the original image as possible.

Step 3. Decoding.
During this stage, the decoding part (decoder) converts the latent vector back to the output image, where the noise will already be suppressed. Thus, after training the model, only the decoder part is required to perform filtering of the new images.
The architecture of the described neural network is presented in Figure 2. In fact, the autoencoder is a generative model and can be used to increase the training samples (Chen and Guo, 2023).

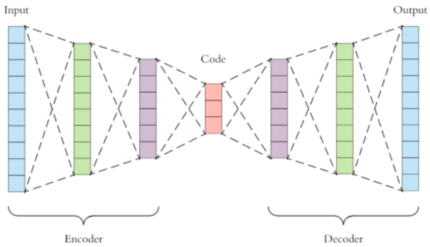


Figure 2. Autoencoder architecture.

Finally, Figure 3 shows an example of noisy images with relative noise variance of 10% and 50%.


Relative Noise Variance is 10%


Relative Noise Variance is 50%

Figure 3. Examples of noisy images.

Figure 3 shows that an image with a relative noise variance of 10% visually looks undistorted, but at a noise level of 50% a strong deterioration of quality is already noticeable. Further we will give the results of autoencoder operation and compare it with other models.

3. Investigation of Image Filtering Efficiency

Let the relative variance of the filtering error will be the main estimation metric. First, let's calculate the absolute variance of the error using the following equation:

$$\sigma_E^2 = \frac{1}{N}\sum_{k=1}^{N}\Big(I_k(i,j,c) - \hat{I}_k(i,j,c)\Big), \qquad (2)$$

where N is number of images for processing, $\hat{I}_k(i,j,c)$ is filtering result of the *k*-th image.

The calculation of the error variance (2) can be used to determine the relative error variance:

$$\delta_E^2 = \frac{1}{N}\sum_{k=1}^{N}\frac{\sigma_{Ek}^2}{\sigma_{Ik}^2}, \qquad (3)$$

where σ_I^2 is image brightness variance, k is image index (number).

It should be noted that efficiency estimation will be produced on 20% test data. And rest 80% of dataset was used for training. We will consider two types of autoencoders. The first one is based on fully connected neural network layers, the second one is based on convolutional layers. Table 2 shows the results of comparison of different models at different noise levels. The relative variance of error is in the table cells depending on model and noise level. Table 3 provides results using SSIM metric.

Algorithm	Noise, 10%	Noise, 20%	Noise, 50%	Noise, 70%	Noise, 100%
Kalman Filter	0.076	0.138	0.372	0.640	0.842
Wiener Filter	0.099	0.194	0.388	0.638	0.826
Median Filter	0.108	0.210	0.340	0.712	0.992
Gaussian Filter	0.104	0.197	0.418	0.695	0.975
Doubly Stochastic Filter	0.082	0.097	0.257	0.571	0.862
Fully Connected AutoEncoder	0.073	0.095	0.262	0.582	0.878
Convolutional AutoEncoder	0.031	0.062	0.199	0.513	0.641

Table 2. Comparison results for different filters (relative MSE)

Algorithm	Noise, 10%	Noise, 20%	Noise, 50%	Noise, 70%	Noise, 100%
Kalman Filter	0.971	0.968	0.903	0.816	0.748
Wiener Filter	0.969	0.963	0.901	0.812	0.752
Median Filter	0.947	0.933	0.844	0.764	0.691
Gaussian Filter	0.951	0.949	0.832	0.712	0.632
Doubly Stochastic Filter	0.971	0.969	0.906	0.822	0.747
Fully Connected AutoEncoder	0.969	0.964	0.912	0.836	0.794
Convolutional AutoEncoder	0.982	0.978	0.958	0.949	0.892

Table 3. Comparison results for different filters (SSIM)

From the presented tables we can conclude that the convolutional autoencoder outperforms the other algorithms. This is due to the complexity of the data and the significant amount of time and computational power for training compared to the Kalman and Wiener algorithms. At the same time, the doubly stochastic filter slightly outperforms the full-link autoencoder due to its universal structure. Furthemore, results for noise level 50% and more are only test results, because they are quite worse and in practice there is no usually such images.

Thus, the results show that convolutional autoencoders exhibit the best noise suppression ability at different noise levels.This is due to the fact that convolutional networks are able to efficiently extract meaningful features from images while suppressing the noise component. However, the fully connected encoder is only ranked third in terms of error metric, which may be due to insufficient training that requires significant computational resources.

https://doi.org/10.5194/isprs-archives-XLVIII-2-W5-2024-9-2024 | © Author(s) 2024. CC BY 4.0 License.

Figure 4 shows examples of noise reduction for autoencoders. On the top is the noisy image, followed by noise suppression by a fully connected model, then noise suppression by a convolutional model is presented third. Figure 5 shows the original image.

Noisy image (25%) | Fully Connected Model | Convolutional Model

Figure 4. Results of denoising.

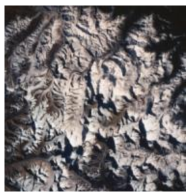


Figure 5. Source image.

Figure 6 shows the effectiveness of filtering on different classes for a convolutional autoencoder on training data.

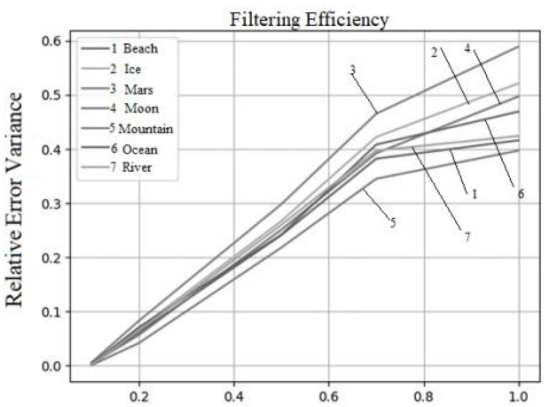


Figure 6. Filtering efficiency by class.

From Figure 6 it is possible to conclude that all encoders are quite effective at low noise levels (up to 0.2), and then the filtering error starts to grow sufficiently. At the same time, the algorithm adapted worst of all to Mars images, which is probably due to its surface.

4. Augmented data processing

The use of data augmentations (Buslaev et al., 2020) is an important aspect of training autoencoders for image filtering tasks. The augmentations allow to expand and diversify the training sample, which helps to improve the generalization ability of the neural network model.

In this study, we proposed the use of different types of augmentations to improve the filtering performance of satellite images. For example, transformations such as random rotations, scaling, shifts and reflections of images have been applied. Such augmentations allow the neural network to better generalize the patterns of noise and artifacts found in real satellite images and improve the filtering quality accordingly.

Experimental results demonstrated that the application of data augmentations significantly improves the performance of the developed autoencoder compared to models trained without the use of image transformations. The authors note that the selection of optimal types and parameters of augmentations is an important step in the design of neural network filters for real-world applications. Competent use of augmentations can significantly improve the efficiency of filtering satellite images using autoencoders.

Figure 7 shows some augmentation examples for noise image of Mars. On the top the source noisy image is represented.

Noisy original | Crop | Rotation

Figure 7. Images obtained using augmentations.

Table 4 shows the results of filtering with for autoencoders with and without augmentation. The noise level is 20% and 50%. Other experiment include test of filters for 20% impulse and white Gaussian noise. Table 5 shows the results for such comparison. The comparison metric is variance of filtering error.

Algorithm	Noise, 20%	Noise, 50%
Fully Connected AutoEncoder	0.095	0.262
Convolutional AutoEncoder	0.062	0.199
Fully Connected AutoEncoder + 10% Augmentation	0.091	0.251
Convolutional AutoEncoder + 10% Augmentations	0.054	0.197
Fully Connected AutoEncoder + 30% Augmentation	0.089	0.252
Convolutional AutoEncoder + 30% Augmentations	0.052	0.196

Table 4. Comparison results with augmenations

Algorithm	Additive White Gaussian Noise	Salt and Pepper
Kalman Filter	0.138	0.204
Wiener Filter	0.194	0.202
Median Filter	0.210	0.067
Gaussian Filter	0.197	0.215
Doubly Stochastic Filter	0.097	0.138
Fully Connected AutoEncoder	0.095	0.154
Convolutional AutoEncoder	0.062	0.112

Table 5. Comparison results for different type of noise

From Table 4 we can see that augmentation provides a reduction in filtering error, but increasing the volume of augmented images from 10 to 30% results in almost no improvement in filtering quality. As for Table 5, the autoencoders was better again besides median filter.

Another interesting case is the processing of aerial images from aircraft, including unmanned aerial vehicles (Knyaz et al., 2024). Distortions in such imagery can be related to the speed of the aircraft and the quality of the cameras used for imaging.

Let us consider the work of the developed algorithm for such images. Figure 8 shows the original image, an observation with a noise variance of 0.7 of the signal variance, and the result of filtering using a convolution autoencoder.

Source

Noisy

Filtered

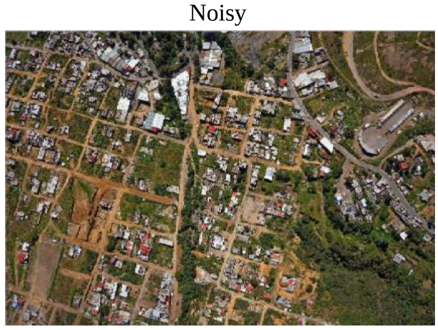
Figure 8. Filtering result of aerial survey data.

Note that the lowest error was obtained using convolutional autoencoder and it amounted to 0.21, while Kalman filtering provided an error variance of 0.4 and Wiener filter provided the error variance of 0.32. Similar studies were conducted for images containing urban buildings.

Figure 9 shows the results of processing such data. The first is original image, second is the noisy image, and then it is possible to see the result of processing using autoencoder and the last is processing using Kalman filter. It should also be noted that in addition to white noise with relative dispersion of 0.7, pixelization distortion was used.

Source

Noisy



Autoencoder

Kalman

Figure 9. Processing of urban development images.

For the presented images, the relative variance of the filtering error is 0.26 in the case of convolutional autoencoder processing and 0.41 for Kalman filter processing. Also, an important result is that the convolutional autoencoder, due to its generative nature, performed much better in dealing with pixelization distortion.

Thus, initial studies on the processing of single aerial images show the high potential of the proposed solution, but further studies on larger amounts of data are required for more accurate evaluations.

5. Conclusions

In this paper, a study on the application of autoencoders for image filtering is carried out. Experimental results showed that the convolutional autoencoder outperforms other known models. At the same time, the second place in terms of error variance was taken by the double stochastic filter. At low noise levels (up to 0.25), a noise variance error of less than 10% can be achieved. The classes of images to be filtered also affect the filtering performance. The best results (for mountains) have 33% less error than the worst result (for Mars). Applying augmentation of noisy images and their clean pairs helps the autoencoders to slightly improve the filtering results, but this is achieved at levels as low as 1%. Result for SSIM metric are quite good too for autoencoder filtering approach, but the using of salt and pepper noise provides worse results in comparison with median filter. In the future, it is planned to test the proposed filtering method on city images and on aerial images.

References

Andriyanov, N.A., Dementiev, V.E., Kargashin, Yu. D., 2021: Analysis of the Impact of Visual Attacks on the Characteristics of Neural Networks in Image Recognition. *Procedia Computer Science.*, 186, 495–502. doi.org/10.1016/j.procs.2021.04.170.

Andriyanov, N.A., Dementiev, V.E., Tashlinsky, A.G., 2022: Development of a Productive Transport Detection System Using Convolutional Neural Networks. *Pattern Recognition Image Analysis*, 32(3), 495–500. doi.org/10.1134/S1054661822030038.

Andriyanov, N.A., Vasil’ev, K.K., Dement’ev, V.E., 2019: Investigation of filtering and objects detection algorithms for a multizone image sequence. *ISPRS Archives*, XLII-2/W12, 7-10. doi.org/10.5194/isprs-archives-XLII-2-W12-7-2019.

Bartlett, O.J., Benoit, D.M., Pimblet, K.A., Simmons, B., Hunt, L., 2023: Noise Reduction in Single-Shot Images Using an Auto-Encoder. *Monthly Notices of the Royal Astronomical Society*, 521(4), 6318–6329. doi.org/10.1093/mnras/stad665.

Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.A., 2020: Albumentations: Fast and Flexible Image Augmentations. *Information* 11, 125. doi.org/10.3390/info11020125.

Chen, S., Guo, W., 2023: Auto-Encoders in Deep Learning—A Review with New Perspectives. *Mathematics*, 11, 1777. doi.org/10.3390/math11081777.

Chen, Z., Wu, H., Li, W., Wang, J., 2023: Enhanced Deconvolution and Denoise Method for Scattering Image Restoration. *Photonics*, 10, 751. doi.org/10.3390/photonics10070751.

Dementyiev, V.E., Andriyanov, N.A., Vasilyiev, K.K., 2020: Use of Images Augmentation and Implementation of Doubly Stochastic Models for Improving Accuracy of Recognition Algorithms Based on Convolutional Neural Networks. *2020 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)*, 1-4, doi.org/10.1109/SYNCHROINFO49631.2020.9166000.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N., 2021: An image is worth 16x16 words: Transformers for image recognition at scale. *Int Conf on Learning Representations*, 1, 1-22.

Kalita, D., Lyakhov, P., 2022: Moving Object Detection Based on a Combination of Kalman Filter and Median Filtering. Big Data. *Cogn. Comput.*, 6, 142. doi.org/10.3390/bdcc6040142.

Kleefeld, A., Breuß, M., Welk, M., Burgeth, B., 2015: Adaptive Filters for Color Images: Median Filtering and Its Extensions. *In: Trémeau, A., Schettini, R., Tominaga, S. (eds) Computational Color Imaging. CCIW 2015. Lecture Notes in Computer Science*, 9016, 14-22. doi.org/10.1007/978-3-319-15979-9_15

Lebedev, M.A., Vizilter, Y.V., Vygolov, O.V., Knyaz, V.A., and Rubis, A.Y., 2018: Change detection in remote sensing images using conditional adversarial networks. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLII-2, 565–571. doi.org/10.5194/isprs-archives-XLII-2-565-2018.

Wang, M., Zheng, S., Li, X., Qin, X., 2014: A new image denoising method based on Gaussian filter. *2014 International Conference on Information Science, Electronics and Electrical Engineering*, 1, 163-167, doi.org/10.1109/InfoSEEE.2014.6948089.

Mursal, A.S.N., Ibrahim, H., 2020: Median Filtering Using First-Order and Second-Order Neighborhood Pixels to Reduce Fixed Value Impulse Noise from Grayscale Digital Images. *Electronic*, 9, 2034. doi.org/10.3390/electronics9122034.

Rubis, A.Yu., Lebedev, M.A., Vizilter, Yu.V., Vygolov, O.V. 2016: Morphological image filtering based on guided contrasting. *Computer Optics*, 40(1), 73-79. doi.org/10.18287/2412-6179-2016-40-1-73-79

Vasiliev, K.K., Dementiev, V.E., Andriyanov, N.A., 2017: Filtration and restoration of satellite images using doubly stochastic random fields. *CEUR Workshop Proceedings*, 1814, 10–20.

Zhu, Y., Jin, G., Liu, T., Zheng, H., Zhang, M., Liang, S., Liu, J., Li, L., 2022: Self-Attention and Convolution Fusion Network for Land Cover Change Detection over a New Data Set in Wenzhou, China. *Remote Sens.*, 14, 5969. doi.org/10.3390/rs14235969.

Knyaz, V.A., Kniaz, V.V., Zheltov, S.Yu., Petrov, K.S., 2024: Multi-sensor Data Analysis for Aerial Image Semantic Segmentation and Vectorization. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLVIII-1-2024, 291–296, doi.org/10.5194/isprs-archives-XLVIII-1-2024-291-2024.