Analysis of Loss Functions for Image Reconstruction Using Convolutional Autoencoder

Nishant Khare, Poornima Singh Thakur, Pritee Khanna(✉), and Aparajita Ojha

Computer Science and Engineering, PDPM IIITDM, Jabalpur, India
{nishantk,poornima,pkhanna,aojha}@iiitdmj.ac.in

Abstract. In recent years, several loss functions have been proposed for the image reconstruction task of convolutional autoencoders (CAEs). In this paper, a performance analysis of a CAE with respect to different loss functions is presented. Quality of reconstruction is analyzed using the mean Square error (MSE), binary cross-entropy (BCE), Sobel, Laplacian, and Focal binary loss functions. To evaluate the performance of different loss functions, a vanilla autoencoder is trained on eight datasets having diversity in terms of application domains, image dimension, color space, and the number of images in the dataset. MSE, peak signal to noise ratio (PSNR), and structural similarity index (SSIM) metrics have been used as the performance measures on all eight datasets. The assessment shows that the MSE loss function outperforms two datasets with a small image dimension and a large number of images. At the same time, BCE excels on six datasets with high image dimensions and a small number of training samples in datasets compared with the Sobel and Laplacian loss functions.

Keywords: Image reconstruction · Autoencoder · Loss functions

1 Introduction

Autoencoders are an unsupervised learning technique, where neural networks are used for representation learning. Recently, autoencoders are being used for learning features [1,2], removing noise (Denoising Autoencoder) [3], image enhancement (deblurring or dehazing images) [4,5], feature retrieval, image coloring, and compression. Autoencoders can reconstruct an image similar to the input image by feature learning and representation and found to be useful for image reconstruction in fields like medical science and forensic science, where the decoded and noise-free images are required from the available incomplete or noisy images [6].

An autoencoder works with encoder and decoder modules. The encoder module learns important features from a given image. The decoder module attempts to reconstruct the matching input image from this representation. The CAE learns useful features to reproduce the input image as the output. Due to the loss of some features, the reconstructed image may not match exactly with the

B. Raman et al. (Eds.): CVIP 2021, CCIS 1568, pp. 338–349, 2022.
https://doi.org/10.1007/978-3-031-11349-9_30

input image. Loss functions play an important role in achieving the desired reconstructed image. The performance of autoencoder depends on input data and the loss function. The motivation behind this work is to explore the performance of a convolutional autoencoder (CAE) using various existing loss functions on datasets from various domains. The objective is to analyze the reconstruction quality of images generated through an autoencoder with various loss functions. A variety of datasets including natural images, sketches, and medical images, are considered to fulfill this objective. The work is organized as follows. Section 2 presents related works. The CAE architecture used in the present analysis is discussed in Sect. 3. Loss functions used for the experimentation are explained in Sect. 4. Experimental setup, including the details of datasets, are discussed in Sect. 5. The experimental results on different datasets in terms of quantitative evaluation and qualitative comparisons are discussed in Sect. 6. Finally, the work is concluded in Sect. 7.

2 Related Work

Recent works on image reconstruction are focused on the use of autoencoders [3, 6,7] (see also, [8,9]). Autoencoders are primarily used for image reconstruction, but have also been utilized in a variety of different tasks like image denoising [3], anomaly detection [7], learning sparse representation and generative modeling. Although convolutional autoencoders can reconstruct images and have noise removal capability, there are specific issues about overall predicted image quality, edges, small artifacts, and blurriness. As autoencoders learn from raw data, they require large number of samples and clean representative data [10]. Also, autoencoders are data-specific and can lead to lossy reconstruction. The amount of acceptable reconstruction loss is application-specific. Considering the impact of the loss layer of neural networks, Zhao et al. [11] compared the performance of several loss functions (MS-SSIM, L1, and SSIM) for image restoration. They showed the importance of perceptually motivated losses when the resulting image is evaluated by a human observer and proposed a novel, differentiable error function. The new metric combines the advantages of L1 and MS-SSIM.

Li et al. [12] proposed steer image synthesis with Laplacian loss. They used neural style transfer based on CNN to synthesizes a new image that retains the high-level structure of a content image, rendered in the low-level texture of a style image. The Laplacian matrix produced by the Laplacian operator helped to detect edges and contours. A new optimization objective for neural style transfer named Lapstyle is achieved using Laplacian loss along with CNN. The minimization of this objective produced a stylized image that better preserves the detail structures of the content image and eliminates the artifacts.

Bai et al. [8] proposed an autoencoder like GAN. The proposed network predicts the missing projections and improves the reconstructed CT (X-Ray) images. Reconstruction loss and adversarial loss are jointly implemented to produce more realistic images. In the generator network, an encoder-decoder is trained to predict missing projections. The output of the decoder and ground

truth for missing projections are fed to the discriminator network for image generation.

Hong et al. [5] proposed an effective loss function that uses the Sobel operator to improve image deblurring methods based on deep learning. They reported that loss functions can affects the network's learning. Compared to conventional deep learning-based image deblurring methods based on the mean square error (MSE) loss function, the Sobel loss function proposed in this work helps the network recover the high-frequency components of resultant images. Zhu et al. [13] proposed a wavelet loss function for the autoencoder to generate the reconstructed image. The wavelet loss function overcomes the problem of blurriness for large dimension images. A new image quality index called wavelet high-frequency signal-to-noise ratio is also proposed to showcase the effectiveness of the wavelet loss function. Most of the previous works using the autoencoder with loss function are focused on the specific type of datasets. The quality of the results may improve significantly with a better loss function with the same network architecture Zhao et al. [11]. A detailed comparison between loss functions is required to differentiate their performance over different datasets. In this paper, we focus on the performance of some frequently used loss functions of different datasets. The contribution in this paper can be summarized as follows:

- Reconstruction performance of a CAE is analyzed with respect to five different loss functions on eight color and grey-scale image datasets from different domains.
- Sobel and Laplacian loss functions are analyzed for their edge reconstruction capabilities in the autoencoder.

3 Convolutional Autoencoder (CAE)

Loss functions plays a crucial role when image reconstruction is performed using a convolutional autoencoder. In this study to analyze the performance of various loss functions in image reconstruction, a simple CAE architecture is chosen. There are total eight blocks in the CAE containing different combinations of convolution layers, downsampling, upsampling, batch normalization, and transpose convolution layers as illustrated in Fig. 1. The convolutional layer performs convolution operations over the images and generates feature maps. These maps store useful features of a given image. The downsampling layers (MaxPool) and the upsampling layers (UpSample) are applied in the convolutional neural network to reduce the dimension and restore the original size of the image, respectively. The batch normalization layer improves training speed by optimizing the model parameters. The transpose convolution layer performs the operations of upsampling and convolution as a single layer. The encoder part performs image feature extraction and the decoder is used to reconstruct input images with the help of encoder features.

The encoder consists of four blocks. Each block has two convolutional layers with ReLU activation function followed by a batch normalization layer. The

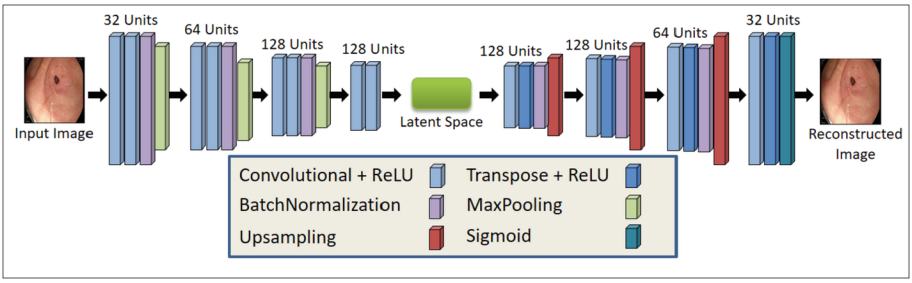

Fig. 1. The proposed CAE network

initial three blocks of the encoder have a max-pooling layer for dimension reduction. In the first block, 32 filters are used in the convolutional layer for feature extraction. In the following blocks, the number of filters is increased by two times, i.e., the second block has 64 filters, the third block has 128 filters, and the fourth block has 128 filters in the convolutional layers. The filter size in the convolutional layers is 3×3, and the maximum pooling size is 2×2. The value of stride is set as 1 in all the blocks. The decoder comprises four blocks in the opposite setup to the encoder part. Each block has a convolutional layer with ReLU activation function and a transpose layer with ReLU activation function followed by the batch normalization layer. The filter size is set as 3×3 in convolution and transpose layer. The decoder part uses transpose convolution to reconstruct the spatial resolution. Transpose convolutions change the order of dimension during convolutional operation. Along with the transpose layer, an upsampling layer is used to retrieve similar-sized feature maps as encoder blocks. An upsampling layer of size 2×2 is used to increase the feature dimensions in the initial three blocks. The first convolutional layer of the decoder takes the input from the last convolutional layer of the encoder, which has 128 feature maps. The decoder's first, second, third, and fourth blocks have 128, 128, 64, and 32 filters, respectively. In the last block, a convolutional layer with three filters is applied with the sigmoid activation function to generate the input image as output. This architecture is selected after multiple rounds of hyperparameter tuning in terms of the number of layers and filters in each layer.

4 Loss Functions

The loss function is used to optimize model performance. It tells us the prediction performance of the model. In the present context, reconstruction loss is a measure to check how good reconstructed images are produced by the proposed autoencoder model. The following loss functions are investigated with the CAE to analyze their suitability for datasets from different domains.

Mean square error (MSE) is the most common regression function which compares the pixels of the input image to the pixels of the reconstructed image. The mean of the corresponding pixel's difference is taken and squared as shown

in Eq. (1) [14]. The advantage of using MSE is to get only one global minimum. There are no local minima. MSE is very easy to compute but very sensitive to outliers. MSE makes error values large by computing the square of them, and the model tries to minimize that cost.

$$MSE = (\frac{1}{mn}) \sum_{i=1}^{n} \sum_{i=1}^{m} (f_{(i,j)} - \hat{f}_{(i,j)})^2 \tag{1}$$

where MSE is the mean squared loss function, f is the true input image fed to the autoencoder, $\hat{f}$ is the reconstructed image by the autoencoder, (i, j) are image pixel location, and $m \times n$ is image dimension.

Binary cross-entropy (BCE) loss compares pixel probabilities of the reconstructed and input image and produces output in terms of 0 or 1. It then calculates the score that penalizes the probabilities based on how close or far the expected value is from the actual value. BCE is the negative average of the log of corrected predicted probabilities as given in Eq. (2) [15]. Sigmoid is the only activation function that is suitable for BCE.

$$\text{BCE} = -(f_i \cdot \log \hat{f}_i + (1 - f_i) \cdot \log (1 - \hat{f}_i)) \tag{2}$$

where $\hat{f}_i$ is the i-th scalar value of the model output, and f_i is the corresponding target value.

Sobel loss function [5,16] is specially designed to highlight the edges of any objects in the input image using the Sobel operator. It gives high-frequency reconstruction guidance with clear and sharp images. Sobel is an edge detection filter that has two kernels, Horizontal and vertical. Convolution of the image with Sobel kernels gives edge maps. The distance between the reconstructed image and input image to the autoencoder can be minimized in terms of minimizing the distance between the two in Sobel edge space. The idea is to calculate the mean squared loss of a Sobel filtered prediction and a sobel filtered ground truth image as shown in Eq. (3) [16,17].

$$SL = (\frac{1}{nm}) \sum_{i=1}^{n} \sum_{j=1}^{m} (F(i,j) - \hat{F}(i,j))^2 \tag{3}$$

where SL is Sobel Loss, F is the Sobel filtered input image fed to the autoencoder, $\hat{F}$ is the Sobel filtered reconstructed image by the autoencoder, (i, j) are image pixel location, and m and n are image dimensions.

Laplacian loss function [12,16] is designed with the Laplacian operator, widely used in image processing and computer vision to detect edges and contours. The Laplacian operator generates images that have gray edge lines and dark background. The Laplacian operator highlights gray level discontinuities in an image and tries to deemphasize regions with slowly varying gray levels. This produces inward and outward edges in an image. Mean squared loss of a Laplacian filtered prediction and a Laplacian filtered ground truth image as given in Eq. (4).

$$LL = (\frac{1}{nm}) \sum_{i=1}^{n} \sum_{j=1}^{m} (G(i,j) - \hat{G}(i,j))^2 \tag{4}$$

where LL is the Laplacian Loss, G is the Laplacian filtered true input image fed to the autoencoder, $\hat{G}$ is the Laplacian filtered reconstructed image by the autoencoder, (i, j) are image pixel location, and m and n are image dimensions.

Focal binary loss function [18,19] generalizes the binary cross-entropy loss by introducing a hyperparameter γ which is a focusing parameter for hard to classify cases. The focal loss [20] is more robust to noisy labels and is defined as in Eq. (5).

$$FL(y, \hat{p}) = -\alpha y(1-\hat{y})^{\gamma} log(\hat{y}) - (1-y)\hat{y}^{\gamma} log(1-\hat{y}) \tag{5}$$

where y is the class label and $\hat{y}$ is the predicted probability of the positive class.

In some works, Structure Similarity measure is also used as the loss function. In this paper, we have used SSIM and peak signal to noise ratio for performance evaluation.

5 Experimental Setup

5.1 Datasets Used for Experimentation and Pre-processing

Experiments have been carried out on 9 datasets selected from three distinct fields. It includes medical images (Kvasir [21], MRI Brain [22], and COVID chest X-Ray [23] datasets), natural images (CIFAR-10 [24], SBU Shadow [25], COIL-100 [26] datasets), and handmade digits and sketches images (MNIST [27] and TU-Berlin [28] datasets) as shown in Table 1. The purpose of selecting multiple datasets from 3 different domains is to analyze the performance of different loss functions on heterogeneous images. The datasets chosen for the experiment have varying image dimensions, and color channels.

Table 1. Datasets for experiment with description

Dataset	Description	Images	Size	Type
TU-Berlin [28]	Handmade sketches	1600	1111×1111	Binary
Kvasir [21]	Medical Images (Endoscopic)	1500	Varying size	RGB
MRI Brain [22]	Brain MRI	253	630×630	Grayscale
SBU Shadow [25]	Shadow images	5938	Varying size	RGB
X-Ray [23]	Covid Chest X-Ray	3000	299×299	Grayscale
COIL-100 [26]	Natural Images	7200	128×128	RGB
CIFAR-10 [24]	Natural Images	60000	32×32	RGB
MNIST [27]	Handwritten digits	70000	28×28	Grayscale

As seen in Table 1, images in various datasets are of different sizes. Moreover, images in SBU Shadow and Kvasir datasets vary in size. For further processing, images of TU-Berlin (1111×1111), Kvasir, MRI Brain (630×630), and SBU

Shadow datasets are resized into 512×512 using interpolation method. Similarly, images of the X-Ray (299×299) dataset are resized into 256×256. In the case of datasets with images of small dimensions, like COIL-100 (128×128), CIFAR-10 (32×32), and MNIST (28×28), the autoencoder with only six blocks is used. For these datasets, the encoder has three blocks with 32, 64, and 128 convolutional filters, and the decoder is mirror of the encoder.

5.2 Evaluation Metrics

The CAE used in this work reduces the size of input images into a smaller representation. There is some information loss while reconstructing the image using the latent representation of the image data. Structural similarity and noise generation during reconstruction by autoencoder are to be observed to find out reconstruction error. Image similarity metrics are used to generate quantitative evaluation between two images. MSE (as discussed in Sect. 4), PSNR, and SSIM are computed to measure the reconstructed image quality.

PSNR (Peak Signal to Noise Ratio) is useful in estimating the efficiency of the autoencoder for representing features. PSNR represents the ratio between the maximum intensity of the image and noise that affects the quality of the image, as shown in Eq. (6) [29].

$$PSNR = 20 \log_{10} \frac{(L-1)}{RMSE} \tag{6}$$

where L is the highest intensity value of the image, and RMSE is the root mean square error.

The SSIM metric compares the structural similarity between two images f the original and the reconstructed images $\hat{f}$. It is a perception-based metric that evaluates image degradation as the error in structural information during reconstruction. Structural information is based on pixel's inter-dependencies that are spatially closed to each other. It is calculated using Eq. (7) [30].

$$SSIM(f, \hat{f}) = \frac{(2\mu_f \mu_{\hat{f}} + t_1) + (2\sigma_{f\hat{f}} + t_2)}{(\mu_f^2 + \mu_{\hat{f}}^2 + t_1)(\sigma_f^2 + \sigma_{\hat{f}}^2 + t_2)} \tag{7}$$

where μ_f, $\mu_{\hat{f}}$, σ_f^2, $\sigma_{\hat{f}}^2$ are the average and variance of f and $\hat{f}$ respectively and $\sigma_{f\hat{f}}$ is the co-variance of f and $\hat{f}$; t_1 and t_2 are constants.

6 Results and Discussion

Both quantitative and qualitative results have been reported in this section. As per the quantitative results shown in Table 2 for all the eight datasets, the MSE and BCE loss functions provide better reconstruction capability to the CAE as compared to Sobel and Laplacian loss functions. The CAE with MSE loss function gives better results for CIFAR-10 and MNIST datasets. However, CAE

with BCE loss outperforms for TU-Berlin sketches, Covid X-Ray, MRI Brain, SBU Shadow, COIL-100, and Kvasir dataets. It can be seen that CAE with MSE loss function performs better for low-dimensional images but with a large number of training samples, whereas CAE with BCE is able to converge well on high-dimension images even with smaller number of images.

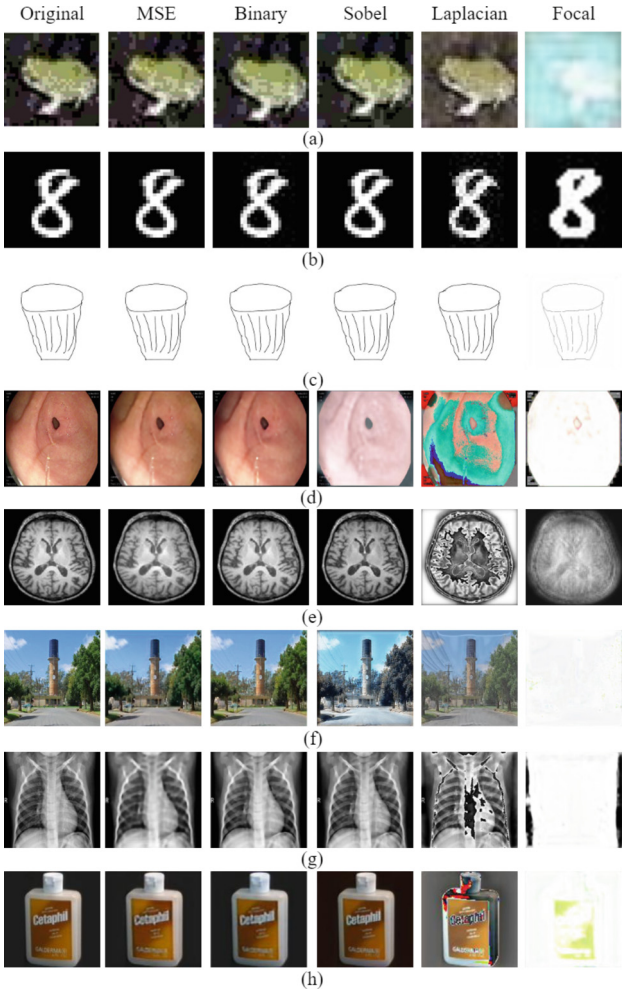

Fig. 2. Visualization of reconstruction results on various datasets: (a) CIFAR-10, (b) MNIST, (c) TU-Berlin, (d) Kvasir, (e) MRI Brain, (f) SBU Shadow, (g) X-Ray, and (h) COIL-100. Loss functions used with CAE are mentioned at the top of the column.

The qualitative results are generated to analyze the reconstruction of edges and lines in terms of image quality. Figure 2 shows the qualitative results for all the datasets. The MSE loss function squares the difference between predicted

Table 2. Quantitative results for all the datasets

Dataset	Loss type	MSE	PSNR	SSIM
CIFAR-10	**MSE**	**30.7353**	**33.5410**	**0.9377**
	Binary	34.8772	32.9695	0.9259
	Sobel	38.3657	32.5157	0.9268
	Laplacian	96.1099	28.3799	0.8819
	Focal	102.9514	0.2604	28.0189
MNIST	**MSE**	**0.0002**	**38.7742**	**0.9983**
	Binary	15.2759	36.5869	0.9380
	Sobel	0.000194	37.8702	0.9985
	Laplacian	32.6411	33.0845	0.7842
	Focal	0.0407	0.7554	62.2592
TU-Berlin	MSE	2.5797	44.4779	0.9900
	Binary	**2.4129**	**44.8066**	**0.9907**
	Sobel	2.5151	44.6088	0.9903
	Laplacian	5.7002	40.7972	0.9778
	Focal	5.0341	0.9603	41.3340
X-Ray	MSE	74.3817	28.6159	0.8735
	Binary	**70.2714**	**30.0993**	**0.8941**
	Sobel	67.7070	30.0734	0.8769
	Laplacian	97.7320	28.2346	0.6269
	Focal	104.0770	0.6093	27.9662
MRI Brain	MSE	25.5430	34.6040	0.9040
	Binary	**22.8570**	**34.7756**	**0.9209**
	Sobel	59.3236	30.7709	0.7860
	Laplacian	108.2985	27.8012	0.4402
	Focal	109.9161	0.2692	27.7253
SBU Shadow	MSE	58.8917	30.5607	0.7249
	Binary	**58.0454**	**30.6898**	**0.7439**
	Sobel	96.8310	28.2956	0.7154
	Laplacian	105.3909	27.9058	0.3136
	Focal	106.2414	0.3950	27.8794
COIL-100	MSE	14.1881	36.9310	0.9523
	Binary	**11.2689**	**38.0655**	**0.9556**
	Sobel	89.4440	28.7077	0.9200
	Laplacian	107.4096	27.8260	0.4455
	Focal	104.5412	0.2859	27.9466
Kvasir	MSE	46.3198	31.6111	0.8534
	Binary	**22.5377**	**34.8506**	**0.8706**
	Sobel	114.2045	27.5561	0.6733
	Laplacian	94.9930	28.3767	0.5526
	Focal	100.3878	0.5715	28.1283

and input images, which makes the error prominent and results in blurry images. In addition, it removes small artifacts, i.e., thin lines and edges are faded from images or are blurred. On the other hand, image reconstruction with BCE shows better performance. The Sobel loss function also works well for producing fine lines and edges. Laplacian loss function generates sharp edges. However, the overall image quality is not good. Also, in the case of the SBU Shadow dataset shown in Fig. 2 (g), it can be seen that the MSE loss function reconstructs blurry electricity cables in the image, whereas BCE and Sobel loss functions perform better than MSE. The predicted image with the Laplacian function is not similar to the original image, as it reconstructs dark and thick electricity cables. The edges and lines in the image generated by the MSE loss function are not as sharp and smooth as the original image in X-Ray and SBU Shadow datasets. The performance of focal loss function is not upto the mark as can be seen from its reconstructed images in Fig. 2 and the quantitative results given in Table 2.

7 Conclusion

In the present work, five different loss functions are evaluated for their strength in enabling a CAE in image reconstruction. The experiments are performed on eight diverse image datasets with MSE, BCE, Sobel, Laplacian and focal loss functions. The results show that the MSE loss function outperforms on datasets with sample low-dimension and large number of training samples. The CAE can adapt well with BCE loss function on high-dimensional images even with a relatively small number of training samples. The CAE with BCE loss function reconstructs good quality images. Sharp and smooth edges of the objects are achieved with CAE using the Sobel loss function. However, slight blurriness is noted in some cases, similar to the MSE loss function results. Overall, the CAE model with binary cross-entropy performs well in most datasets, whereas it is close to the MSE loss function in some experiments.

References

1. Odaibo, S.: Tutorial: deriving the standard variational autoencoder (VAE) loss function. arXiv preprint arXiv:1907.08956 (2019)
2. Kingma, D.P., Welling, M.: An introduction to variational autoencoders. arXiv preprint arXiv:1906.02691 (2019)
3. Gondara, L.: Medical image denoising using convolutional denoising autoencoders. In: 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), pp. 241–246. IEEE (2016)
4. Fazlali, H., Shirani, S., McDonald, M., Brown, D., Kirubarajan, T.: Aerial image dehazing using a deep convolutional autoencoder. Multimedia Tools Appl. **79**(39), 29493–29511 (2020)
5. Hong, J.-P., Cho, S.-J., Lee, J., Ji, S.-W., Ko, S.-J.: Single image deblurring based on auxiliary Sobel loss function. In: 2020 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia), pp. 1–3. IEEE (2020)

6. Chen, M., Shi, X., Zhang, Y., Wu, D., Guizani, M.: Deep features learning for medical image analysis with convolutional autoencoder neural network. IEEE Trans. Big Data **7**, 750–758 (2017)
7. Chow, J.K., Su, Z., Wu, J., Tan, P.S., Mao, X., Wang, Y.-H.: Anomaly detection of defects on concrete structures with the convolutional autoencoder. Adv. Eng. Inform. **45**, 101105 (2020)
8. Bai, J., Dai, X., Wu, Q., Xie, L.: Limited-view CT reconstruction based on autoencoder-like generative adversarial networks with joint loss. In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 5570–5574. IEEE (2018)
9. Liu, X., Gherbi, A., Wei, Z., Li, W., Cheriet, M.: Multispectral image reconstruction from color images using enhanced variational autoencoder and generative adversarial network. IEEE Access **9**, 1666–1679 (2020)
10. Pandey, R.K., Saha, N., Karmakar, S., Ramakrishnan, A.G.: MSCE: an edge-preserving robust loss function for improving super-resolution algorithms. In: Cheng, L., Leung, A.C.S., Ozawa, S. (eds.) ICONIP 2018. LNCS, vol. 11306, pp. 566–575. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04224-0_49
11. Zhao, H., Gallo, O., Frosio, I., Kautz, J.: Loss functions for image restoration with neural networks. IEEE Trans. Comput. imaging **3**(1), 47–57 (2016)
12. Li, S., Xu, X., Nie, L., Chua, T.-S.: Laplacian-steered neural style transfer. In: Proceedings of the 25th ACM International Conference on Multimedia, pp. 1716–1724 (2017)
13. Zhu, Q., Wang, H., Zhang, R.: Wavelet loss function for auto-encoder. IEEE Access **9**, 27101–27108 (2021)
14. Ephraim, Y., Malah, D.: Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. IEEE Trans. Acoust. Speech Signal Process. **33**(2), 443–445 (1985)
15. Creswell, A., Arulkumaran, K., Bharath, A.A.: On denoising autoencoders trained to minimise binary cross-entropy. arXiv preprint arXiv:1708.08487 (2017)
16. Lu, Z., Chen, Y.: Single image super resolution based on a modified U-net with mixed gradient loss. arXiv preprint arXiv:1911.09428 (2019)
17. Kanopoulos, N., Vasanthavada, N., Baker, R.L.: Design of an image edge detection filter using the Sobel operator. IEEE J. Solid State Circuits **23**(2), 358–367 (1988)
18. Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), October 2017
19. Wang, C., Deng, C., Wang, S.: Imbalance-XGBoost: leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost. Pattern Recognit. Lett. **136**, 190–197 (2020)
20. Ma, X., Huang, H., Wang, Y., Romano, S., Erfani, S., Bailey, J.: Normalized loss functions for deep learning with noisy labels. In: International Conference on Machine Learning, pp. 6543–6553. PMLR (2020)
21. Pogorelov, K., et al.: KVASIR: a multi-class image dataset for computer aided gastrointestinal disease detection. In: Proceedings of the 8th ACM on Multimedia Systems Conference, pp. 164–169 (2017)
22. Chakrabarty, N.: Brain MRI images for brain tumor detection. https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection (2019)
23. Rahman, T., et al.: Exploring the effect of image enhancement techniques on COVID-19 detection using chest X-ray images. Comput. Biol. Med. **132**, 104319 (2021)

24. Krizhevsky, A., Nair, V., Hinton, G.: CIFAR-10 (Canadian institute for advanced research), vol. 5, vol. 4 (2010). http://www.cs.toronto.edu/kriz/cifar.html
25. Vicente, T.F.Y., Hou, L., Yu, C.-P., Hoai, M., Samaras, D.: Large-scale training of shadow detectors with noisily-annotated shadow examples. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 816–832. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_49
26. Nene, S.A., Nayar, S.K., Murase, H., et al.: Columbia object image library (COIL-100) (1996)
27. LeCun, Y., Cortes, C.: MNIST handwritten digit database (2010). http://yann.lecun.com/exdb/mnist/
28. Eitz, M., Hays, J., Alexa, M.: How do humans sketch objects? ACM Trans. Graph. (Proc. SIGGRAPH) **31**(4), 44:1–44:10 (2012)
29. Silva, E.A., Panetta, K., Agaian, S.S.: Quantifying image similarity using measure of enhancement by entropy. In: Mobile Multimedia/Image Processing for Military and Security Applications 2007, vol. 6579, p. 65790U. International Society for Optics and Photonics (2007)
30. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)