DOI : https://doi.org/10.56452/7-4-242

ISSN: 0974-5823 Vol. 7 No. 4 April, 2022

International Journal of Mechanical Engineering

Design and analysis of hybrid model by use of the latest deep learning techniques (CAE) for loss compression

Sandeep Chand Kumain

Asst. Professor, Department of Computer Science, Graphic Era Hill University, Dehradun, Uttarakhand India 248002

Abstract:

Compressed data travel fastly through internet during communication and consume less size and space. So Image compression is very important in this internet era and this can be achieved easily by deep learning approach. One of the module through which image compression can be achieved is CAE which stands for convolutional auto encoder technique. In this article to compress the image and calculate the images size and color concentration, we discuss a multiple layer hybrid system which is developed by unsupervised CAE structure of deep learning and for color assembling it uses k-mean algorithm. In this article KODAK and CLIC datasets are used to compare and test the image compression.

Keywords: CLIC, KODAK, Deep learning, CAE, unsupervised.

1. Introduction

As people have limited capacity or storage to save the images, videos and data, so image compression technique is the hot topic for the researcher in recent years. The compression can be applied on any of the media whether it is video, audio, text, graphics, images, animation etc. Image compression can applied to any type of image format. This technique is used to reduce the memory space so that it can be easily and speedy transfer from one system to another and this ultimately reduce the transmission cost. Image size reduction is performed by omitting the redundancy in the images. Mainly two image compression approaches are used, one is lossy and another is lossless compression. To reconstruct the original input image, the unsupervised Auto encoder deep learning technique with clustering technique is used. Its main goal is to copy its input to its output, in other words, it attempts to reconstruct the original input image, and also considered clustering is one of the widely used methods. There are so many methods for clustering the data like K-Means which is an unsupervised, numerical, easiest, and fast method to compute cluster from given data. In this article, we discuss a hybrid system, in which clustering features are added to k mean algo and CAE technique.

DOI : https://doi.org/10.56452/7-4-242

2. Material and Methodology

2.1 Datasets: KODAK and CLIC datasets are used to compare and test the image compression.

2.2 Autoencoder

An autoencoder is a type of artificial neural network that encodes data. An autoencoder's goal is to learn anew representation of the input data, either compressed or sparse. An autoencoder is a type of unsupervised learning system in which the target values are set to be equal to the input values, yi = x. (i). As seen in Figure (2) an autoencoder usually consists of one input layer, one or two or more hiddenlayers, and one output layer with the same number of units as the input layer. In certain cases, different layers of the network add a sequence of non-linear transformations to input data. Different representationsof the input data are used by the hidden layers [14]. During the training process, an autoencoder has two functional components: an encoder and a decoder. The encoder converts the input data to the desired lossy format[15].

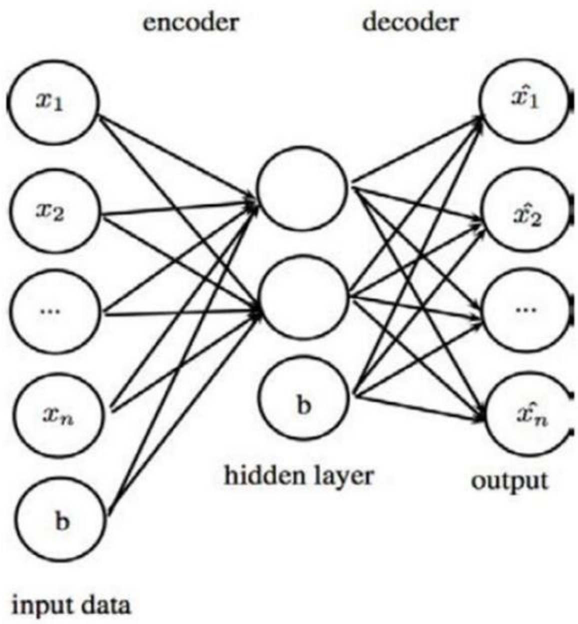

Figure 2 : Simple 3-layer AutoEncoder structure [45].

Compressed (or sparse) representation by applying transformations hj (the j layer), while the decoder decodes this compressed representation to an approximation of the inputs $X^{\wedge(i)}$, with $X^{\wedge(i)}$ as close to X as possible, In the training phase of an autoencoder, minimized the error such that minimized the average reconstruction. The reconstruction error is used to measure the similarity between $x^{\wedge(i)}$ and $x^{(i)}$which can be measured in different ways. the squared error Are simple but commonly used, for any input $x^{(i)}$ [15].

$$\mathbf{L}\ (\ ^{(i)}) = \sum_{k=1}^{d} (x_k^{(i)} - x_k^{\wedge(i)})^2 \quad \text{(Eq.1)} \ [15]$$

where $x^{(i)}$ is a d-dimensional vector and (k) is a hidden unit.in this research, The Convolutional Neural Network (CNN) algorithm is usually applied to train an autoencoder (CAE) we used to compress RGB image

DOI : https://doi.org/10.56452/7-4-242

5 . Autoencoder Deep learning Network

in the proposed hybrid image compression system is applied and implementation of the CAE using the Convolutional Neural Network (CNN) algorithm, to compress image data, and the CAE consists of several layers of different sizes that connect with each other to form the CAE network (Adam) type, and each layer is responsible for a specific operation within the network, The dataset in the Autoencoder network goes through two stages:

A. Encoder (Compressing)

The first stage of the CAE network is the image encoder (compression) stage which consists of several layers, the CLIC and Kodak dataset is entered into the input layer (Normalization layer), where the dataset size is referred to as a mathematical representation Y=H*W*C, the number of value is C=3 for the R and G and B channels (H) represents the row and (W) column. The compression stage consists of the following layers which are uniformly arranged within the CAE network the first layer convolutional layer, which is responsible for applying the 3×3 size filters as 16 different filters were used in the AutoEncoder network to obtain the best results in the compression stage and to maintain the compressed image quality. The activation layer or (RELU activation function), which is responsible for the compression output of compressed images with a high level of accuracy. The function of the activation layer (RELU) in the CAE network and other neural networks is to determine the output of the neural network to obtain high-accuracy results. Down-sampling or(max-pooling) , the first and second layers follow a group of layers called a (down-sampling layers) or also called the (max-pooling layer), which works to take the largest values from the compressed image matrix while preserving the image information (features), the filter size 3×3 is used in the down-sampling layer

B. Decoder (Decompressing)

The second stage of the CAE network is the decoder (decompression) and dataset (images) retrieval stage, and it is considered the most important stage because it is responsible for recovering compressed data (images), the Normalization layer works in the decompression stage at restoring the representation of the resulting images (the input images) in a smaller size and high quality. to represent the images resulting from the compression process and restore them repeatedly The up-sampling layer (max pooling) in the decompression process leads to obtaining the highest rates of representation of the good characteristics of the resulting images with high accuracy. Figure (3) shows the compression (encoder) and decompression (decoder) phases of the automatic encoder network and the representation of the layer.

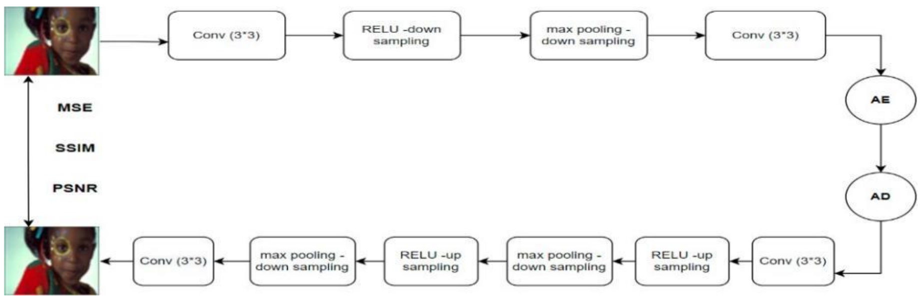

Figure 3: proposed system of CAE network compression image

DOI : https://doi.org/10.56452/7-4-242

6. Hybrid System

To implement the model, the algorithm (k-mean) with CAE by CNN was used to design a hybrid image compression system. Figure (4) shows the stages of compression (encoder) and decompression (decoder) that the system and images go through during the training and testing phases. the dataset They pass through many layers of CAE to be processed during the input and output process input and outputprocess.

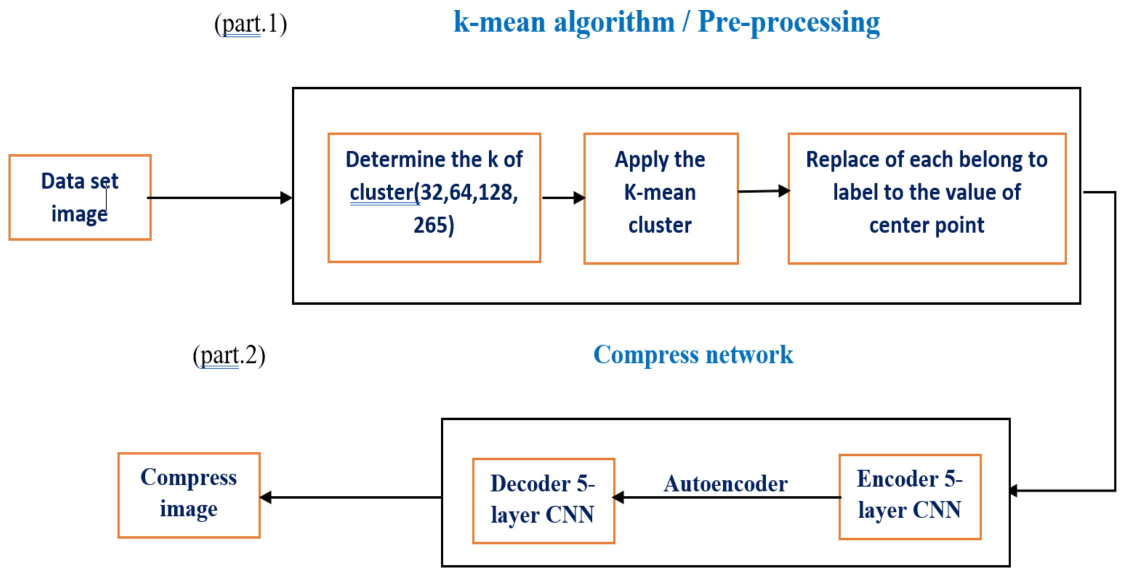

Figure 4: Proposed System of k-mean Hybrid Compression Image

7. Analysis Deep Learning Model

In the first stage of our proposed system, data images are entered into the k-mean algorithm, which the images pass through in three stages:

A. The first stage is in which the images are converted from a 2-dimensional matrix to a 1-D vector to make it easier for the system to read and deal with them.

B. The second stage the number of clusters to be used in the system is determined, we used 64 clusters in our system and it is possible to use the following clusters 32, 64,128, 265 instead of them, and after determining the clusters, the algorithm converts the two-dimensional matrix into a one-dimensional matrix or vector that needs for work, and a group of center points is specified for it that selects the pixels near each A point from the center points until 64 clusters are formed.

C. In the second stage, each value of the totals is replaced by the Centre point to which it belongs and is replaced by the matrix to result in a matrix of a smaller image that can be distinguished by the abstract eye.

8. Analysis Result

Each test image of a dataset (Kodak and CLIC) is evaluated using a hybrid model of the proposed system displays the compression results of the model when applying the K-Mean and CAE algorithm. Tables (1) and (2) show the effectiveness of all parameters used in the proposed system and their effect on the quality of the resulting compressed (Encoder) and reconstructed (Decoder) image. The test results are evaluated and compared using the following accuracy standards (MSE, PSNR, SSIM) pressure rating as shown in the tables

DOI : https://doi.org/10.56452/7-4-242

below.

Table 1: Result of compression (kodak dataset test)

n.of (k) cluster	Metric Parameter			Compression
	PSNR	SSIM	MSE	RATE
2	66.3620	0.9843	0.0156	0.8707
4	68.3447	0.9899	0.0100	0.7793
8	69.1314	0.9916	0.00839	0.7375
16	68.6332	0.9903	0.0096	0.7227
32	69.8333	0.9925	0.0074	0.7274
64	69.5497	0.9920	0.0079	0.7128
128	70.8003	0.9938	0.0061	0.7117

Table 2: Result of compression (CLIC dataset test)

n.of (k) cluster	Metric Parameter			Compression
	PSNR	SSIM	MSE	RATE
2	67.5479	0.9873	0.0126	0.9930
4	72.7477	0.9960	0.0039	0.9880
8	77.0917	0.9984	0.00151	0.9777
16	80.8600	0.9993	0.0006	0.9285
32	84.1956	0.9996	0.0003	0.8661
64	87.0921	0.9998	0.00015	0.7771
128	70.1807	0.9938	0.0067	0.7191

Experimental Results of CAE Image compression steps mages are compressed using the automatic CAE that was built using the CNN algorithm and consists of several layers, These layers work on compressing images and selecting the best features for them depending on The value of the clustering parameter(k) of the dataset images resulting from the compression in the first stage, which is entered into the autoencoder as input for the second stage, where that dataset goes through two stages, (i): compression (encoder) to obtain compressed images. (ii): decompression (decoder) to retrieve the original images that have been compressed using the autoencoder, which is similar to the original images entered in the same degree of colors to them and the intensity of degree of color and quality of the image varies each time a different value is entered for the clustering parameter (k) for the first stage. When reducing the value of the clustering parameter (k) between (2-4-8), the system works to choose the least number of color pixels for the image matrix in each cluster of the compressed image (decoder) resulting

DOI : https://doi.org/10.56452/7-4-242

from the CAE so that the color accuracy and quality are weak compared to the original images, but when the parameter value (k) increases between (16-32-64-128) the system works to choose the largest number of color pixels for the image matrix in each cluster of the compressed image resulting from the autoencoder, so that the color accuracy and quality of the retrieved compressed images (decoding) are very high and are close to the original images entered with high Compression Ratios (CR). Consequently, it increases the values of the accuracy parameters of the resulting compressed images and also reduces the value of the errorcoefficient (MSE) for them, The results in tables (1) and (2) the effect of changing the clustering factor

(k) on the final compression results for test samples of image experiments for datasets of the Kodak andCLIC datasets for the proposed hybrid system.

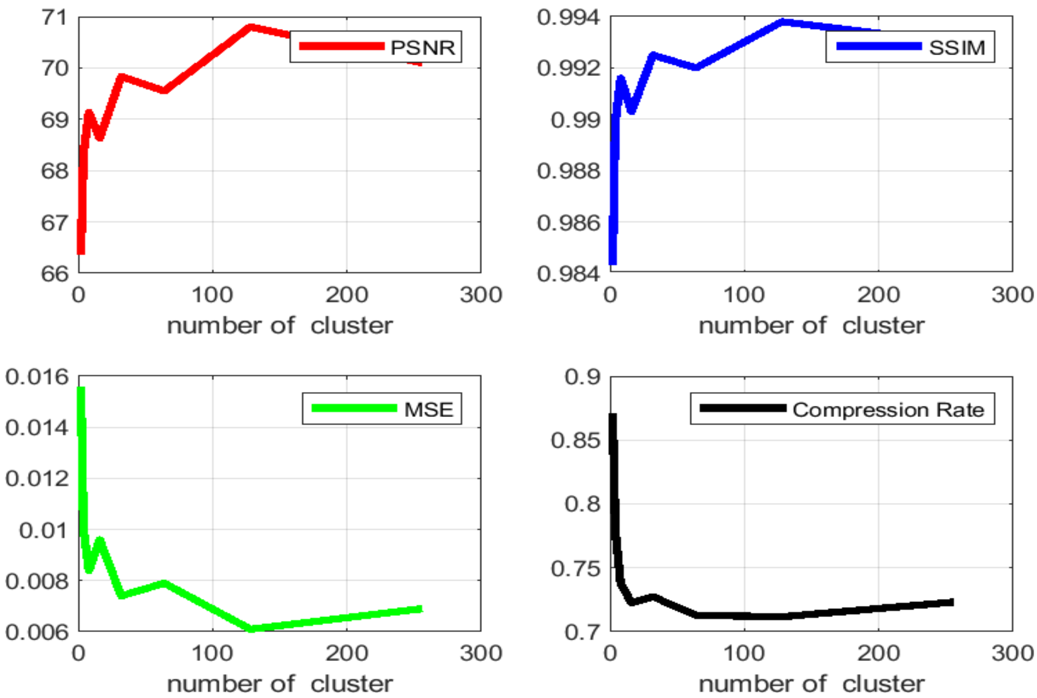

Figure 5: result test for clustering parameter effect

DOI : https://doi.org/10.56452/7-4-242

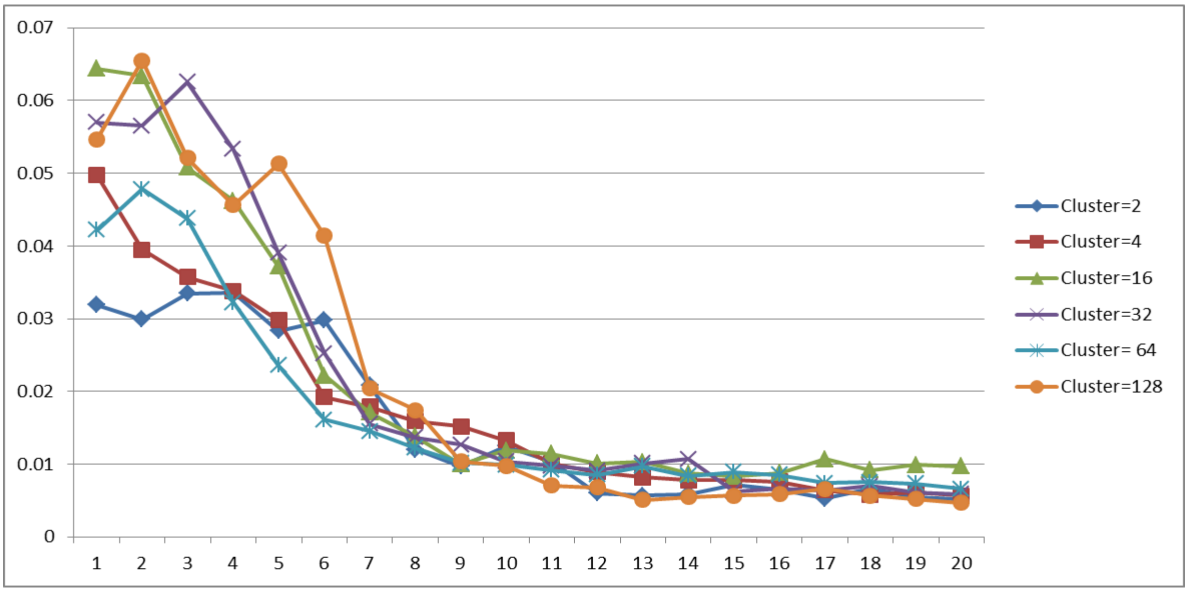

Figure 6: result test for clustering parameter of compression rate (CR)

as well as the graphs (4) and (5) present the process of increasing the compression ratio (CR) as the value of the clustering factor increases in the system. The final results of the compressed images of the Kodak and CLIC datasets are shown in the table (3) below, the effect of the clustering factor on the image qualityof the CAE each time.

Table 3: Final compression result for kodak and CLIC dataset with changing cluster

Kodak Dataset		**CLIC Dataset**	
	Original image Size 598 KB		Original image Size 3.38 MB
	Cluster=2 Size 85.6 KB		Cluster=2 Size 100 KB
	Cluster=4 Size 131 KB		Cluster=4 Size 220 KB

DOI : https://doi.org/10.56452/7-4-242

	Cluster=8 Size 169 KB		Cluster=8 Size 338 KB
	Cluster=16 Size 175 KB		Cluster=16 Size 493 KB
	Cluster=32 Size 177 KB		Cluster=32 Size 742 KB
	Cluster=64 Size 179 KB		Cluster=64 Size 1.04 MB
	Cluster=128 Size 181 KB		Cluster=128 Size 1.51 MB

Conclusion

In this article, we discuss a hybrid model based on the use of the latest deep learning techniques (CAE) for loss compression, and we employed the clustering technique of the K-mean algorithm to determine thevolume of the dataset that the system compresses for multiple images. In a hybrid way with the k-mean algorithm to improve the performance of the CAE and increase the speed of the system to compress the largest number of images for the Kodak and CLIC dataset and with the help of the autoencoder layers for encoder and decoder, It improves the visual quality of the image. In addition to the system, we suggest adding a loss determination function (loss function) that determines the value of the loss to train it and achieve higher results using other deep learning techniques in addition to the type of autoencoder.

Reference

[1] G. Verma and A. Kumar, “Image Compression Using Deep Learning Based Multi-structure Feature Map and K-Means Clustering,” *Commun. Comput. Inf. Sci.*, vol. 1240 CCIS, pp. 365–374,2020, doi: 10.1007/978-981-15-6315-7_30.

[2] Erich Schubert, Jorg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu.2017. DBSCAN Revisited ,Revisited: Why and How You Should (Still) Use DBSCAN.ACM Trans.Database Syst. 42, 3, Article 19 (July 2017), 21 pages

[3] V. N. V. Satya Prakash, K. Satya Prasad, and T. Jaya Chandra Prasad, “Color image demosaicing using sparse based radial basis function network,” *Alexandria Eng. J.*, vol. 56, no. 4, pp. 477–483, 2017, doi: 10.1016/j.aej.2016.08.032.

[4] X. Wu, “Color demosaicking by local directional interpolation and nonlocal adaptive thresholding,” *J. Electron. Imaging*, vol. 20, no. 2, p. 023016, 2011, doi: 10.1117/1.3600632.

[5] K. R. Deshmukh, “Image Compression Using Neural Networks,” 2019.

[6] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, “Deep Convolutional AutoEncoder-based Lossy Image Compression,” *2018 Pict. Coding Symp. PCS 2018 - Proc.*, pp. 253–257, 2018, doi:10.1109/PCS.2018.8456308.

[7] Y. Zhang, “A Better Autoencoder for Image: Convolutional Autoencoder,” pp. 1–7, 2015.

[8] Y. Zhang, “A Better Autoencoder for Image: Convolutional Autoencoder,” pp. 1–7, 2015.

[9] S. Zebang and K. Sei-Ichiro, “Densely connected autoencoders for image compression,” *ACM Int. Conf. Proceeding Ser.*, vol. Part F1477, pp. 78–83, 2019, doi: 10.1145/3313950.3313965.

[10] S. Zebang and K. Sei-Ichiro, “Densely connected autoencoders for image compression,” *ACM Int. Conf. Proceeding Ser.*, vol. Part F1477, pp. 78–83, 2019, doi: 10.1145/3313950.3313965.

[11] S. Zebang and K. Sei-Ichiro, “Densely connected autoencoders for image compression,” *ACM Int. Conf. Proceeding Ser.*, vol. Part F1477, pp. 78–83, 2019, doi: 10.1145/3313950.3313965.