Date of publication SEP-30, 2024, date of current version AUG-02, 2024.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.23.3.3660

Unsupervised Representation Learning using Wasserstein Generative Adversarial Network

IFTAKHER HASAN MOHAMMED TAREK[1], MOHAMMED MAHMUDUR RAHMAN[1], ZULKIFLY MOHD ZAKI[2]

[1]Department of Computer Science and Engineering, International Islamic University Chittagong-4318, Bangladesh
(e-mail: mdhasan867656@gmail.com, mmr.cse@iiuc.ac.bd)
[2]Faculty of Science and Technology, University Sains Islam Malaysia, 71800 Nilai, Negeri Sembilan, Malaysia
(e-mail: zulkifly@usim.edu.my)

Corresponding author: Iftakher Hasan Mohammad Tarek (e-mail: mdhasan867656@gmail.com)

ABSTRACT In recent years, representational learning has attracted considerable attention. However, unsupervised representation learning has received less attention compared to supervised representation learning. This paper introduces a combination of a deep neural network (DNN) and a generative adversarial network (GAN) that can learn features through unsupervised learning. Essentially, the Generative Adversarial Network (GAN) is a deep learning architecture that engages two neural networks in a framework similar to a zero-sum game. Generating new, synthetic data that resembles a known data distribution is the aim of GANs. In June 2014, Ian Goodfellow and associates first developed the idea of Generative Adversarial Network (GAN). The research used a new type of GAN model which is called Wasserstein GAN. There are some distinct differences between traditional GAN and Wasserstein GAN. This paper highlights the differences and benefits of using Wasserstein GAN, as well as the architecture of Wasserstein GAN. This study trained the model on an image dataset to extract features, and subsequently tested it on another dataset, demonstrating that the GAN model learns a hierarchy of representation from object parts in the discriminator. The purpose of this paper is to use unsupervised learning like Convolutional Neural Network (CNN) and Wasserstein Generative Adversarial Network (WGAN) for feature extraction from unlabeled dataset.

KEYWORDS Deep Neural Network; Generative Adversarial Network; Wasserstein GAN, Convolutional Neural Network; Unsupervised Learning; Representation Learning.

I. INTRODUCTION

LEARNING features from large amounts of images is an active research topic. Computer vision models need to learn features from images in different kinds of tasks, such as image classification, image colorization, or object direction. It is essential for many applications across many industries, such as computer vision, healthcare, security, and entertainment. It makes it possible for computers to comprehend and interpret visual data, which opens up a wide range of creative and useful use cases. In various types of deep learning tasks, whether supervised or unsupervised, learning features from an image dataset is crucial. The image datasets are mostly unlabeled.

Deep learning techniques have a transformative impact across diverse domains, including artistic creation and medical diagnostics, in the rapidly evolving field of artificial intelligence. Research and development activities in deep learning are dynamic and expected to drive further advancements in the field. The goal is to create AI systems that are more powerful, effective, and ethically sound. This trajectory is highly influenced by the complex needs arising from various application domains, demonstrating a dedication to customizing AI solutions to particular contextual problems.

Generative Adversarial Networks (GANs) are one of the many methods leading this evolution that has attracted significant interest. GANs, distinguished by their unique approach of harnessing the antagonistic interaction between the discriminator and the generator, two neural networks, have proven their usefulness in a diverse array of applications. Notably, they differ significantly from traditional machine learning models, which frequently rely on carefully labeled datasets, in that they can learn meaningful features from unlabeled, raw data.

Typically, a large labeled data corpus is required to achieve visually appealing results in feature learning. However, this

work's adopted paradigm, unsupervised learning, offers a more efficient approach to extract meaningful patterns from unlabeled data, marking a significant step towards real machine autonomy. The study under review suggests a novel combination of convolutional neural networks (CNNs) and generative adversarial networks (GANs), which together provide a potent mechanism for feature extraction from unlabeled data. The resulting model is
positioned at the nexus of the supervised and unsupervised learning paradigms and is known as the Deep Convolutional Generative Adversarial Network (DCGAN).

There aren't many studies investigating the possibilities of GAN models, especially when it comes to unlabeled datasets, which emphasizes how novel and important this work is. The suggested methodology adds a significant component to the expanding body of knowledge in the field of deep learning by using the discriminator or generator as a means of representation learning for supervised tasks. This study contributes to the existing body of knowledge on GANs in the context of unlabeled data, paving the way for future advancements in the field.

II. RELATED WORK

The Generative Adversarial Networks (GANs) model has not received much research. On the other hand unsupervised representation learning is a well studied problem in computer science research. This research used unsupervised representation learning with the GAN model.

In [1] this paper, a new GAN was introduced which is called LAPGAN. The author develops a low resolution generated image which can be reliable. But this model failed to learn features from unlabeled images. Some methods like [2, 3] are based on the Convolutional Neural Network (CNN) and worked very well learning features from images. They even achieved state-of-art results. The drawback of these methods is that models need enormous labeled data to obtain the results.

Certain generative models, like Autoencoders (AE), have been proposed to reconstruct images and create unsupervised learning representations that can extract features from the input and produce a latent space representation [4, 5] that is far more accurate. Even though the reconstruction error that is used to train them might not be the best metric for learning representation in particular, the AE's training process is entirely unsupervised.

Clustering the data (e.g., with K-means) and using the clusters to boost classification scores is a traditional method of unsupervised representation learning. Hierarchical clustering of image patches can be used to learn effective image representations in the context of images [6].

In [7] this paper context demonstrated how to determine the approximate function of each convolution filter in the network by employing deconvolutions and filtering the maximal activations. Likewise, by applying a gradient descent to the inputs, [8] can examine the optimal image that triggers specific subsets of filters.

One of the most well-liked unsupervised algorithms in recent years [9] is the idea of generative adversarial networks (GANs), which offer an engaging framework for two networks to play a two-player minimax game to learn data distribution from unlabeled datasets. They are appealing to representation learning because of their unsupervised learning process. But one of the main issues with GANs is that their training phase is unstable, which frequently leads to generators that produce meaningless outputs that are dispersed throughout the training data.

There are two types of generative image models that have been thoroughly examined: parametric and non-parametric.

Non-parametric models are frequently employed in texture synthesis [10], super-resolution [11], and in-painting [12]. They match from a database of pre-existing images, frequently matching patches of images.

In early 2000, [13] models generated textual images using MNIST dataset. However, generating neutral images has succeeded most recently. Deconvolution network approach [14] and a recurrent approach [15] had got success in generating neutral images. But they have not leveraged for supervised learning.

Most recently, Deep Convolutional GAN (DCGAN) [16] was introduced which also gives state of the art results and leverages for supervised tasks. But the only problem of this model is model collapse. The model can learn its distribution partially and generate just some.

In this paper, the research solves the model collapse problem by introducing Wasserstein GAN. This Gan uses Earth Mover (EM) distance instead of a traditional matrix that solves the vanishing gradient problem. The goal of this paper is to develop a Wasserstein Generative Adversarial Networks (GAN) model with a convolutional neural network (CNN) that will learn features from unlabeled data without model collapse and can leverage supervised learning.

The proposed mode will be sufficiently effective to resolve the model collapse issue that most GAN models previously encountered.

III. METHODOLOGY

One well-known class of artificial intelligence algorithms that is frequently used in the machine learning and deep learning domains is called Generative Adversarial Networks (GANs). This paper conducts a thorough analysis of the suggested model, the Wasserstein GAN (WGAN), in the section that follows. The explanation explores the WGAN's nuances, explaining what makes it unique and clearly separating it from the traditional GAN architecture. This section attempts to provide a refined understanding of the innovations and distinctive features that set the Wasserstein GAN architecture apart from the conventional GAN paradigm by offering a thorough analysis of the architecture.

A. WASSERSTEIN GAN ARCHITECTURE

The Wasserstein Generative Adversarial Network (WGAN), sometimes known as the WGAN, is a variation on the regular GAN that was developed to overcome some of the stability and convergence problems that traditional GANs have. As like traditional GANs the objective of Wasserstein GAN (WGAN) is to make the critic accurately distinguish between real and created data, while the generator's goal is to provide data that is as real as feasible. This phase is called min max game. This game is between the generator which produces new data or images and the discriminator which distinguishes between real and fake data. As figure 1 shows, a noise z is used in the generator to produce synthetic data samples [17]. The produced fake image then processes in the discriminator (critic) alongside the real image to produce the loss value. The main difference between Wasserstein GAN and Traditional GAN is in Discriminator (Critic). There are differences in distance matrix and loss function.

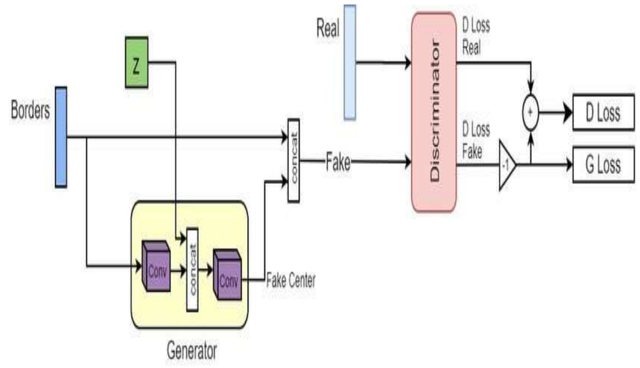


Figure 1. Architecture of Wasserstein GAN

B. WASSERSTEIN DISTANCE

The goal of the WGAN generator is the same as that of the traditional GAN generator. However, the distance metric in WGAN and traditional GAN is different. The generator's goal is to produce samples that closely mimic the actual data distribution from random noise. The generator's objective is to deceive critics into thinking that the generated data is accurate.

So, the objective function is designed to minimize the Wasserstein distance between the real and generated data distributions. The Wasserstein distance is a key difference between traditional gan and WGAN which is a more stable metric than Jensen-Shannon divergence or Kullback-Leibler divergence employed in traditional GANs . The Earth-Mover's distance is another name for the Wasserstein distance, it estimates the "cost" of changing one distribution into another. Therefore, ***G*** seeks to minimize a function that evaluates ***D***'s ability to tell the difference between the real and fake:

$$\frac{min}{G}\,\mathbb{E}_{z\sim p_z(z)}\big[D\big(G(z)\big)\big]. \tag{1}$$

In equation (1), **D** is referred to as the "critic" in this context because it is trained to approximate the Wasserstein distance between real and produced distributions rather than to categorize data. **G** is the generator that produces fake samples from the noise **z**.

The critic in a WGAN seeks to calculate the Wasserstein distance between the real and generated data distributions. In the traditional gan, the discriminator (in traditional gan the critic called discriminator) tries to determine whether data is real or fake whereas in WGAN's critic try to calculate the distance between real and fake data. The critic tries to maximize a function of the wasserstein distance between the average value of real image and the average value of generated/fake images:

$$\frac{max}{D}\,(\mathbb{E}_{x\sim p_{real}(x)}[D(x)]\ -\mathbb{E}_{z\sim p_z(z)}[D(G(z))]) \tag{2}$$

In equation (2), $\mathbb{E}$ represents the expected value, $p_{real}(x)$is the distribution of the real data, and $p_z(z)$ is the prior distribution for generating fake data. **G** is the generator that produces fake samples from the noise **z**.

After maximizing the objective function, the critic ***D*** creates a meaningful gradient that the generator ***G*** can use to advance itself. So generator ***G*** tries to minimize its objective function and the critic ***D*** tries to maximize its objective function. It is called a min-max game.

C. WASSERSTEIN LOSS

There is a major difference in the use of loss function between traditional GAN and Wasserstein GAN. In traditional GANs usually use binary cross-entropy whereas in WGAN use Wasserstein loss. For binary classification issues, the binary cross-entropy loss is frequently used. The discriminator in a GAN is simply a binary classifier because it seeks to distinguish between real and generated (false) samples. Cross-entropy is a measure of the difference of two probability distributions. In binary cross-entropy GAN uses labels 0 and 1. Label 0 is for fake images and label 1 for real images.

Discriminator loss:

$$loss_D = -\mathbb{E}_{x\sim p_{real}(x)}[log\ D(X)] - \mathbb{E}_{z\sim p_z(z)}[log\ (1 - D(G(z)))] \tag{3}$$

Generator loss:

$$loss_G\ = -\mathbb{E}_{z\sim p_z(z)}[log\ D(G(z))] \tag{4}$$

However, the problem of binary cross-entropy is that when the model runs for a while it collapses or the gradient descent does not update and the value becomes constant. It is called the vanishing gradient problem. In traditional GANs, the generator may encounter vanishing gradients, particularly if the discriminator grows too powerful. This happens as a result of the binary cross-entropy loss overload that can produce small gradients in typical GANs. The generator's ability to learn can be hampered by small gradients. Without taking into account the diversity of the actual data distribution, the generator begins to produce a little variety or perhaps just one sort of output. Without actually understanding the underlying data distribution, the generator sort of discovers a "shortcut" to deceive the discriminator. And this mode of generator is called 'Mode Collapse'.

Our proposed model uses the Wasserstein loss to solve this problem. We design the Wasserstein loss by minimizing the Wasserstein distance, also known as the Earth Mover distance. WGAN, unlike traditional GAN, uses any label value for both real and fake images. Therefore, the value is not limited to 0 and 1. That is what solves the mode collapse.

Wasserstein Critic loss:

$$loss_{Critic}\ = \mathbb{E}_{x\sim p_{real}(x)}[C(x)]\ - \mathbb{E}_{z\sim p_z(z)}[C(G(z))] \tag{5}$$

Wasserstein Generator loss:

$$loss_{generator}\ = \mathbb{E}_{z\sim p_z(z)}[C(G(z))] \tag{6}$$

For Wasserstein loss (WLoss) there is a condition that must enforce in the size of the critic's gradients that it should be limited in size. This condition is called 1-Lipschitz continuous condition. Itt means the norm of the gradient should be 1 or less at each point. This is important for stable training with WLoss and to approximate Earth mover's distance correctly. There are

2 ways to enforce the 1-Lipschitz continuous condition which are Weight Clipping and Gradient Penalty. In the proposed model, this research uses Gradient Penalty.

Gradient Penalty is a term added as a regularization term to the loss, to ensure the critic is 1-L continuous. So there is a new term going to add to the loss equation:

$$x = \alpha * real + (1 - \alpha) * fake \tag{7}$$

$$gp = (||\nabla c(x)||_2 - 1)^2 \tag{8}$$

In equation (8), *gp is calculated* by applying an interpolated mix of real and fake images (because it's impossible to check the gradient at every point of the feature space). So the final Wasserstein critics loss equation is:

$$loss_{Critic} = \mathbb{E}_{x \sim p_{real}(x)}[C(x)] - \mathbb{E}_{z \sim p_z(z)}[C(G(z))] + \lambda(||\nabla c(x)||_2 - 1)^2 \tag{9}$$

In equation (9), λ is intensity term of the gradient penalty, controls the strength of the penalty.

IV. TRAINING & TESTING

In this study, the model trained with Imagenet-1k [18]. The pre-trained model is used to generate images from CIFAR-10 [19] dataset and use the images in the L2-SVM classifier. So the model never trained with CIFAR-10 [19] dataset but still generated good images and the study used the L2-SVM to find how good the WGAN is in representation learning.

A. MODEL SETUP

This study distinguishes the model's architectural arrangement by including five layers in both the generators and critics. The overall goal of the research, to enable the training of generative models with higher resolution and depth, justifies this design decision. This quest, through extensive model exploration, resulted in the identification of a structured family of architectures that performed well during the dataset's training.

Three explicit architectural principles are followed in the instantiation of the convolutional Wasserstein Generative Adversarial Network (WGAN). First, batch normalization is incorporated into both the discriminator and the generator [20]. This method, which achieves a zero mean and unit variance by normalizing the input of each unit, is used to stabilize the learning process. In light of the possible difficulties resulting from insufficient initialization, batch normalization stands out as a prominent technique that resolves training problems and improves gradient flow, particularly when dealing with deeper models.

Second, in terms of the generator, the hyperbolic tangent (Tanh) activation is used in the output layer, while Rectified Linear Unit (ReLU) is the activation function used in all other layers [21]. It has been empirically observed that this deliberate use of activation functions accelerates the learning trajectory of the model, enabling a faster saturation and coverage of the color space that is intrinsic to the training distribution. The bounded activation is known for its effectiveness in obtaining faster convergence, especially in the form of Tanh.

Thirdly, all layers of the critics use the Leaky Rectified Linear Unit (LeakyReLU) activation [22]. This decision shows prowess, especially in higher-resolution modeling, and is especially pertinent to the discriminator's context. The discriminator becomes more sensitive and skilled at identifying subtleties in the input data when non-linearity is carefully introduced, especially in the negative domain.

In conclusion, the convolutional WGAN demonstrates a wise and empirically supported methodology through its carefully thought-out architecture, informed by the concepts of batch normalization and careful activation function selection. This method aims to make training more complex and high-resolution generative models more consistent and effective. It is based on a systematic study of architectural arrangements and makes a big contribution to the larger discussion on advanced generative adversarial networks.

During the model's training regimen, the Adam optimizer [23] is employed to facilitate the optimisation process. Its parameters include a momentum parameter of 0.5 and a learning rate of 0.0002. This optimizer improves training process efficiency and convergence and is well-known for its effectiveness in stochastic optimisation tasks.

Moreover, the critics specifically use the Leaky Rectified Linear Unit (Leaky ReLU) activation function [22]. Here, a critical factor controlling the level of leakiness in the activation function, the parameter alpha, is set to 0.2. The addition of controlled non-linearity by the Leaky ReLU, especially in the negative domain, enhances the critics' perceptual abilities and helps the model identify subtle characteristics in the input data.

In order to optimize the model's training dynamics, a careful and well-informed decision went into choosing the optimizer and activation function, as well as fine-tuning the related hyperparameters. Under the convolutional Wasserstein Generative Adversarial Network (WGAN) architecture used in this study, such careful configuration is essential to achieving convergence, stability, and efficient representation learning.

B. DATASET

The research at hand used the CIFAR-10 [19] dataset as its main source of data for model training and evaluation. CIFAR-10 consists of 60,000 color images with a 32 x 32 pixel resolution that are methodically arranged into 10 different classes. The dataset is further divided into subsets, with 10,000 images set aside for model performance testing and 5,000 images designated for training. It is standard procedure in machine learning assessments to separate datasets into training and test sets in order to gauge the model's capacity for generalization. The dataset is useful as a benchmark for image classification tasks because it contains a wide range of visual content that is contained within the designated classes.

Figure 2, which shows representative samples from this image repository, has been included to give a visual insight into the nature of the CIFAR-10 dataset. This graphical depiction aims to provide a concrete comprehension of the visual variety and intricacy contained in the CIFAR-10 dataset, highlighting its importance in supporting thorough model training and assessment in the context of image classification.

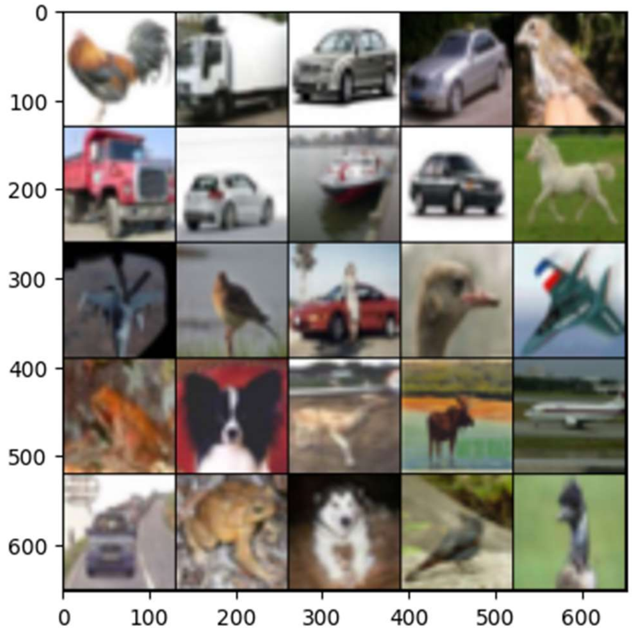


Figure 2. Data sample of CIFAR-10 Dataset

C. RESULT & DISCUSSION

The research uses the Imagenet-1k dataset [18] as a source of real images for unsupervised training. During the training process, 32 × 32 min-resized center crops are used. The main goal is to evaluate the Wasserstein Generative Adversarial Networks (WGAN) representation learning performance. The model is trained using the Imagenet-1k dataset to determine how well the learned representations perform in supervised tasks. Figure 3 provides a visual representation of the evaluation of representation learning performance by showing the evolution of the generator loss and critic's loss over the dataset's training. This graphical representation sheds light on the convergence and effectiveness of the unsupervised learning process by showing how the WGAN model adjusts and fine-tunes its parameters during training.

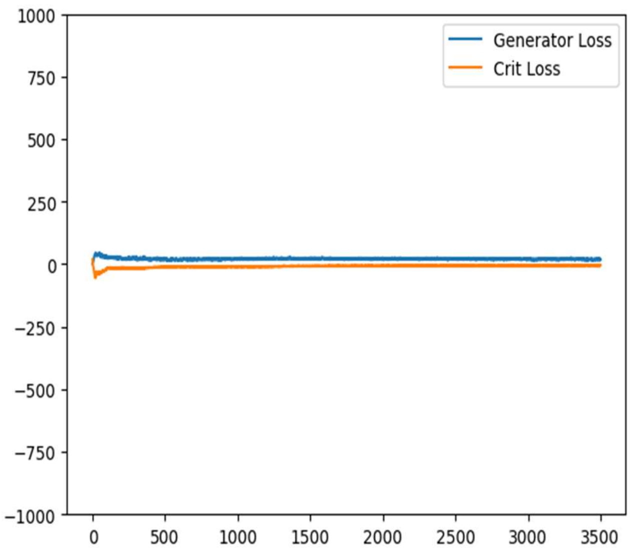


Figure 3. Generator and Critics loss

It then uses CIFAR 10 dataset in the convolutional features from all layers of the critics, max pooling each layer's representation to create a 4 × 4 spatial grid. Following their flattening and concatenation into a 28672 dimensional vector, these features are then used to train a regularized linear L2-SVM classifier. In the process, The CIFAR 10 split into 2 parts; training dataset and test dataset. The pre-trained WGAN model is used to train and test the L2-SVM classifier. first the model train with training dataset. To measure the performance, the test dataset is used to test the accuracy of the model and it gives an accuracy of how the WGAN performs for representation learning. So basically, WGAN never trained with the CIFAR-10 dataset but it trained with Imagenet-1k dataset. However, the pre-trained WGAN used to generate images from CIFAR-10 to train l2-SVM classifier.

This achieved 83.2 %, which is better than other Deep convolutional GANs. The table 1 shows the different results of the CIFAR-10 dataset used in different models..

Applying unsupervised representation learning algorithms as a feature extractor on supervised datasets and assessing the effectiveness of linear models fitted on top of these features is a popular method for assessing the quality of these algorithms. A well-tuned single-layer feature extraction pipeline using K-means as the feature learning algorithm has shown very strong baseline performance on the CIFAR-10 dataset. This technique achieves 80.6% accuracy when using 4800 feature maps, which is an extremely large number. The base algorithm's unsupervised multi-layered extension achieves 82.0% accuracy [24]. Using traditional GAN and deep convolutional neural networks (DCGAN) [16] CIFAR-10 dataset achieved 82.4 %. In that contreset this research's model deep convolutional wasserstein gan has achieved 83.2 % which is better then most of the studies.

But Still, The result is not as good as Exemplar CNN [25], whose accuracy is 84.3%. This paper leaves it for future work to increase the accuracy.

Table 1: Accuracy on the CIFAR-10 dataset.

Model	Accuracy	Accuracy (400 per class)	Max # of features units
1 Layer K-mean	80.6 %	63.7 %	4800
3 Layer K-means Learned RF	82.0 %	70.7 %	3200
View Invariant K-means	81.9 %	72.6 %	6400
Exemplar CNN	84.3%	77.4 %	1024
DCGAN	82.8%	73.8 %	512
WGAN (This research)	83.2 %	74.1 %	512

V. CONCLUSION

This paper presents the Wasserstein Generative Adversarial Networks (WGAN) as a useful model for unsupervised representation learning. A key aspect of this research is a thorough explanation of the differences between the architectures of conventional Generative Adversarial Networks (GAN) and the WGAN paradigm. The empirical results presented in this paper demonstrate that WGAN achieves an industry-leading level of accuracy, especially when it comes to extracting significant features from datasets that lack explicit labeling.

A significant aspect of this work is the methodical way in which it tackles a gap that is widely present in the literature: models' vulnerability to instability and possible collapse over long training intervals. The empirical data presented here confirms that WGAN is effective in addressing these issues, which increases its relevance in the larger context of unsupervised representation learning techniques.

While the recorded achievements are commendable, the scientific investigation remains cognizant of the need for scientific advancement. Therefore, we acknowledge the existing possibility of improving accuracy metrics. This concession also suggests future directions for future research projects, shedding light on the ongoing scholarly effort to improve and enhance unsupervised representation learning techniques.

References

[1] E. L. Denton, S. Chintala, A. Szlam, R. Fergus, "Deep generative image models using a Laplacian pyramid of adversarial networks," ArXiv abs/1506.05751, 2015.

[2] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller and T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 9, pp. 1734-1747, 2016, https://doi.org/10.1109/TPAMI.2015.2496141.

[3] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Proceedings of the 25th International Conference on Neural Information Processing Systems*, Curran Associates Inc.: Lake Tahoe, Nevada, vol. 1, 2012, pp. 1097-1105.

[4] P. Vincent, et al., "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res*., vol. 11, pp. 3371-3408, 2010.

[5] Z. Zhu, et al., "Deep learning representation using autoencoder for 3D shape retrieval," *Neurocomputing*, vol. 204, pp. 41-50, 2016. https://doi.org/10.1016/j.neucom.2015.08.127.

[6] A. Coates, A. Y. Ng, "Learning feature representations with k-means," In Neural Networks: Tricks of the Trade, pp. 561–580. Springer, 2012. https://doi.org/10.1007/978-3-642-35289-8_30.

[7] M. D. Zeiler, R. Fergus, "Visualizing and understanding convolutional networks," Proceedings of the International Conference on Computer Vision (ECCV'2014), Springer, 2014, pp. 818–833. https://doi.org/10.1007/978-3-319-10590-1_53.

[8] A. Mordvintsev, C. Olah, M. Tyka, "Inceptionism: Going deeper into neural networks," 2015. [Online]. Available at: http://googleresearch.blogspot.com/2015/06/inceptionism-going-deeper-into-neural.html.

[9] I. J. Goodfellow, et al., *Generative Adversarial Networks*, 2014.

[10] A. Efros, T. K. Leung, et al., "Texture synthesis by non-parametric sampling," In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999, volume 2, pp. 1033–1038. https://doi.org/10.1109/ICCV.1999.790383.

[11] W. T. Jones, R. Thouis, E. C. Pasztor, "Example-based super-resolution," *Computer Graphics and Applications IEEE*, vol. 22, issue 2, pp. 56–65, 2002. https://doi.org/10.1109/38.988747.

[12] J. Hays, and A. A. Efros, "Scene completion using millions of photographs," *ACM Transactions on Graphics (TOG)*, vol. 26, issue 3, p. 4, 2007. https://doi.org/10.1145/1276377.1276382.

[13] J. Portilla, E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *International Journal of Computer Vision*, vol. 40, issue 1, pp. 49–70, 2000.

[14] A. Dosovitskiy, J. T. Springenberg, T. Brox, "Learning to generate chairs with convolutional neural networks," arXiv preprint arXiv:1411.5928, 2014. https://doi.org/10.1109/CVPR.2015.7298761.

[15] K. Gregor, I. Danihelka, A. Graves, D. Wierstra, "Draw: A recurrent neural network for image generation," arXiv preprint arXiv:1502.04623, 2015.

[16] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," CoRR, 2015. abs/1511.06434

[17] P. Ebner, A. Eltelt, Audio inpainting with generative adversarial network, 2020.

[18] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, F.-F. Li, "ImageNet: A large-scale hierarchical image database," *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 2009, pp. 248-255, https://doi.org/10.1109/CVPR.2009.5206848.

[19] A. Krizhevsky, "Learning multiple layers of features from tiny images," pp. 32-33, 2009.

[20] S. Ioffe, C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," arXiv preprint arXiv:1502.03167, 2015.

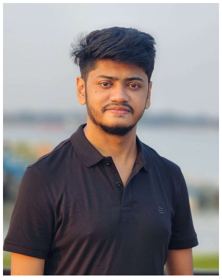
[21] V. Nair, G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[22] B. Xu, et al., "Empirical evaluation of rectified activations in convolutional network," ArXiv e-prints, 2015.

[23] D. P. Kingma, J. Ba, "Adam: A Method for Stochastic Optimization," CoRR, 2014. abs/1412.6980.

[24] A. Coates, A. Y. Ng, "Selecting receptive fields in deep networks," Proceedings of the 24th International Conference on Neural Information Processing Systems (NIPS'11), Curran Associates Inc., Red Hook, NY, USA, 2011, 2528–2536.

[25] A. Dosovitskiy, P. Fischer, J. T. Springenberg, M. Riedmiller, T. Brox, "Discriminative unsupervised feature learning with exemplar convolutional neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, issue 9, pp. 1734–1747, 2016. https://doi.org/10.1109/TPAMI.2015.2496141.



IFTAKHER HASAN MOHAMMED TAREK, is a graduate from faculty of Computer Science and Engineering, International Islamic University Chittagong (IIUC), Tarek researches artificial intelligence and machine learning.

MOHAMMED MAHMUDUR RAHMAN, currently works at the Faculty of Computer Science and Engineering, International Islamic University Chittagong (IIUC), as Associate Professor. Rahman researches pattern recognition, computing, system design and ontology.

ZULKIFLY MOHD ZAKI, currently works at the Faculty of Science and Technology, University Sains Islam Malaysia (USIM), as Associate Professor. Zulkifly's research focuses on Software Design & Development, Human Computer Interaction, Semantic Web, Computer & Usability and User Interaction within a system..