International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

Random-Shaped Image Inpainting using Dilated Convolution

Nermin M. Salem, Hani M. K. Mahdi, and Hazem M. Abbas

Abstract*: Over the past few years, *Deep learning-based methods have shown encouraging and inspiring results for one of the most complex tasks of computer vision and image processing; Image Inpainting. The difficulty of image inpainting is derived from its' need to fully and deeply understand of the structure and texture of images for producing accurate and visibly plausible results especially for the cases of inpainting a relatively larger region. Deep learning methods usually employ convolution neural network (CNN) for processing and analyzing images using filters that consider all image pixels as valid ones and usually use the mean value to substitute the missing pixels. This result in artifacts and blurry inpainted regions inconsistent with the rest of the image. In this paper, a new novel-based method is proposed for image inpainting of random-shaped missing regions with variable size and arbitrary locations across the image. We employed the use of dilated convolutions for composing multiscale context information without any loss in resolution as well as including a modification mask step after each convolution operation. The proposed method also includes a global discriminator that also considers the scale of patches as well as the whole image. The global discriminator is responsible for capturing local continuity of images texture as well as the overall global images' features. The performance of the proposed method is evaluated using two datasets (Places2 and Paris Street View). Also, a comparison with the recent state-of-the-art is preformed to demonstrate and prove the effectiveness of our model in both qualitative and quantitative evaluations.

***Index Terms*: Image inpainting, gan, l1 , dilated convolution.**

I. INTRODUCTION

Image inpainting often refers to the process of filling a missing region in an image. Also, it can be used for the removal of unwanted objects and filling these areas from the surrounding neighborhood of images. The main challenge facing image inpainting is the quality of the produced images with respect to the human eye. Even though it is easier to inpaint smaller regions, inpainting larger regions remains a challenging problem due to its high variabilities and complexities in the image's texture's space.

Many previous deep learning methods depend on Convolution Neural Network (CNN). They considered the inpainting of a square missing region either with a fixed size usually located at the center of images [1,2], or with variable size scattered at any random location of images [3,4]. The work in this paper presents a global image inpainting model. This model should be able to inpaint any random-shaped missing regions with high-level understanding and recognition for the structure and texture of the image. The output of the model should be an accurate prediction of the missing region consistent with the rest of the image without any headache of any post-processing or blending techniques. CNN-based models often learn to minimize a predefined loss function such as L1 or L2 reconstruction loss. The objective of these loss functions is to define the quality of the output results, but these losses often tend to produce blurry outputs as they depend on measuring the Euclidean distance between predicted and ground truth pixels [1,5]. This can be justified by the fact that minimizing Euclidean distance is performed by averaging all plausible outputs.The latest methods of image inpainting employed a CNN along with Generative Adversarial Network (GAN). The pioneer of that field was proposed in [1], and the training model of GAN is based on a comparison between two images; the real one versus the inpainted generated output. GAN consists of two main networks: Generator and discriminator. The discriminator is used to differentiate whether the inpainted generated image is real or fake according to a computed GAN loss during training. Afterwards, it became an inspiration and a motivation for many following recent work [6-9]. They employed encoder-decoder architecture for efficient feature learning and generative adversarial networks for the generation of more coherent results by encouraging the similarity between the generated and real images.In [1], GAN was used for small images, however it could not work for higher-resolution images with random-shaped missing areas. In [10], PatchGAN was presented which updated the setting of the discriminator to work at the scale of patches. This modified discriminator will be able to evaluate whether each $N \times N$ patch in an image is real or fake. CNN-based methods usually suffer from two main problems. The first problem is the use of standard convolutions' filters which may fail to model distant correlation between the distant influencing region and the missing region.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
Nermin M. Salem*, Computer and Systems Engineering Department, Ain Shams University, Cairo, Egypt.
Electrical Engineering Department, Future University, Cairo, Egypt.
Hani M. K. Mahdi, Computer and Systems Engineering Department, Ain Shams University, Cairo, Egypt.
Hazem M. Abbas, Computer and Systems Engineering Department, Ain Shams University, Cairo, Egypt.

The second problem is that the standard filter always substitutes the missing content with a normalized fixed value, i.e. mean value as used in [11].The proposed model overcomes these two problems by using dilated convolutions for the generator's inpainting network to obtain an increased area for each inpainting layer; giving a wider field of view without any increase in computation cost of standard convolutions, and in addition, a modification mask step is associated with each layer. Considering a binary mask, the modification step is concerned with the removal of any mask pixels where the convolution was able to operate on a valid pixel value. The idea is to produce an accurate independent prediction for a new missing binary mask in each step regardless of the initial missing binary mask of the image. Therefore, any random-shaped mask will eventually fade away, i.e. shrink, using the right corresponding number of layers. Also, the use of a single global discriminator is proposed and designed to be suitable for the random regions while maintaining stable and fast training.

In summary, we present in this work:

- A novel-based method capable of inpainting of any random-shaped missing region.
- Employment of the use of dilated convolutions associated with a modification mask step for inpainting network.
- The use of a single patch-based global discriminator suitable for random shaped masks.
- Analysis of each used network component.
- Performance evaluation of the proposed model using two different datasets, i.e. Places2 [12] and Paris Street View dataset [13].
- Qualitative and quantitative analysis with recent tate-of-the-art inpainting methods.

This paper is structured as follows. Related work is provided in Section 2. Approach is briefly explained in Section 3. Experiments and Results are discussed in Section 4. Conclusions are provided in Section 5 and References are presented in Section 6.

II. Related Work

Existing image inpainting methods can be categorized into three main categories: Diffusion-based, Patch-based and Deep Learning-based.

Diffusion-based Methods

These were the earlier methods trying to find a solution for the image inpainting problem. Their methods used image propagation content from the surrounding region for inpainting. Different methodologies were presented for image propagation content such as [14,15], the authors used isophote direction field for propagation. The authors in [16] used a histogram for statistics from the image itself for obtaining local contents. These methods were generally used for inpainting of small regions such as scratches in images.

Patch-based Methods

These methods can inpaint a larger missing region than what diffusion-based methods can. They work on the similarity between patches for borrowing information from the surrounding pixels around the region to be inpainted. These methods were initially targeting texture synthesis [17,18], they worked on a bidirectional patch-similarity based on the dense computation to determine similarity between patches. This method was not practical for applications due to its expensive computation. In [19], The authors presented a modified version based on image energy optimization, they computed the outputs' quality image by using Markov Random Field (MRF)-based model. The results were good for images with similar textures. Another modified approach was presented in [20], it is a fast algorithm for finding and selecting the nearest neighbors for filling the missing hole, this method showed good results for editing image applications. Patch-based methods can be used for filling relatively large regions in a simple structure natural image but did not show any good results for images with complex structures. Also, these methods were not considered fast enough to cope with the modern real-time applications.

Deep Learning-based Methods

Recent Deep learning-based systems often employed GAN for the automatic production of realistic images without any human interference. However, training a GAN network is a hard task due to the instability in the discriminator's behavior which affects generator's convergence. Another issue, the GAN grades the whole images to be real or fake according to an adversarial loss computed by the discriminator network. This method is more suitable for lower resolution images. However, higher resolution images need more attention for local details as well as the global structure. Various GAN architectures were proposed during the last few years to solve the mentioned issues [10, 21-24].Deep learning-based methods and GAN-based methods usually use an initialization for the missing regions/holes with fixed value, usually the mean value [11] which is propagated through the convolution neural network afterwards. In [1], they trained a deep neural network for inpainting of a center-square mask of size 64×64 in 128×128 images with a function of two measure losses, i.e. L2 reconstruction loss and adversarial loss. The results were good for center-region inpainting but failed to produce the same results for any other hole's location. In [2], a two-step technique was proposed; in the first step, the model from [1] is used for initial inpainting hole, afterwards, they applied a global texture model with convolutional neural networks [25] for fine tuning. This work only produced good inpainting results for symmetric images around the center.In [7], they modified the work of [1] by using two discriminators - local and global - the local discriminator is responsible for ensuring local consistency of the inpainted region with the near neighbors, while the global discriminator is responsible for ensuring the image's overall coherence. This work focuses on inpainting objects with similar known structures such as faces. The authors in [6] modified the work in [7] and added a contextual layer for finer tuning of inpainted images. The results were visibly good images, but these techniques can only inpaint square-random missing holes, which is considered as a limitation from our perspective.

Recent PatchGAN proposed in [10,26] presents a simple yet effective network that has the flexibility to adapt to different missing regions' structures. Since it grades each image's patch rather than grading the whole image. The goal of PatchGAN is to let the discriminator runs in a different manner than the traditional GAN, i.e. to classify if each $N \times N$ image patch is real or not. This discriminator runs convolutionally across the image, averaging all responses to provide the ultimate output of D. Our work overcomes the ineffectiveness and limitations of the existing methods by the capability of inpainting diverse scenes. The proposed work can inpaint any random-shaped missing regions at any arbitrary locations across the image.

III. Approach

We build our model by modifying and reproducing the most recent state-of-the-art image-inpainting model presented in [6] and PatchGAN presented in [10]. We were motivated by the encouraging results shown in inpainting of square-masked regions in natural images and faces. In this section the steps will be as following; the used network architecture, convolution and mask modification step, and finally the used loss function will be discussed.

Network Architecture

The overall inpainting network presented in this paper is composed of two main networks: generator inpainting network and a single global discriminator as shown in Fig.1.

Generator Inpainting Network: The generator inpainting network is based on a fully convolutional network. The generator network takes an input pair composed of the ground truth image and the binary mask to indicate the missing regions as an initial input to produce the corrupted image. This corrupted image is then fed to the network inpainted image production.For the generator inpainting network, we employed an encoder-decoder architecture. Using this architecture gives us the advantages of reducing both the memory usage and the computation time, as initially, the corrupted image's resolution is decreased before any processing and then it is recovered to its original resolution at the decoder side using up/sampling or deconvolution layers [27]. Some recent work [28] employed the use of U-net [29] architecture but it is found to be inaccurate in cases of using wider masks as stated in [30].

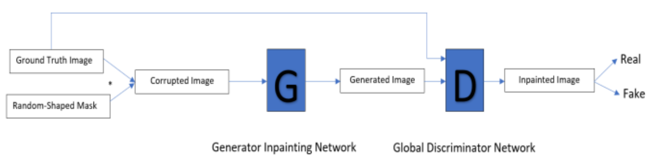

Fig. 1. Used Network Architecture. The generator inpainting network is trained only with $L1_{rec}$ loss. The global discriminator network is trained with $L1_{rec}$ and GAN loss.

A fully dilated convolution also called atrous convolutions [31] is used for all layers of generator inpainting network. Dilated convolution is considered as a modified convolution operation, where it uses a larger kernel coverage, i.e. giving a global view, allowing capturing of more contextual information without any increase of parameters or computation complexity and still maintaining the same used resolution as shown in Fig. 2. It also provides a faster training with a smaller number of parameters.The choice of using dilated convolutions gives the model an important advantage of seeing a wider area of the input image especially for lower resolution layer than the standard convolution. The standard convolution is considered as a dilated convolution with dilation factor D equals to 1.All training images used are of size 256×256. For the encoder part, Rectified Linear Unit (ReLU) activation function is used, and leaky ReLU with α=0.2 is used for the decoder part instead of using Exponential Linear Unit (ELU) used by [6]. A detailed architecture is shown in Table 1a.

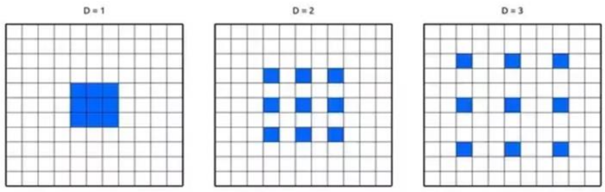

Fig.2. The effect of using dilation convolution for the same kernel size 3×3 but with different dilation factor D=1,2 and 3.

Global Discriminator Network: The goal of this network is to decide whether the generated image from the inpainting network is real or not, i.e. fake. Our generator inpainting network improves its generation through the discriminators' GAN loss also known as Adversarial loss.The global discriminator network is composed of four layers and one fully connected layer. All convolution layers are associated with a stride = 2 in order to decrease the image resolution while increasing the output filters' number. The recent work for weight normalization is used in [32] known as spectral normalization which provides stability for discriminator's training to produce a better adequate quality of the generated images.

Table 1a. Generator Inpainting Network Architecture. After each dilated convolution layer, except the last one, there is a Rectified Linear Unit (ReLU) layer. The output layer consists of a convolutional layer with a clipping function to limit the output to the [0, 1] range. The Outputs column refers to the number of output channels for the output of the layer.

Type	Kernel	Dilation	Stride	Outputs
Conv.	5x5	1	1x1	32
Conv.	3x3	1	2x2	64
Conv.	3x3	1	1x1	64
Conv.	3x3	1	2x2	128
Conv.	3x3	1	1x1	128
Conv.	3x3	1	1x1	128
Dilated conv.	3x3	2	1x1	128
Dilated conv.	3x3	4	1x1	128
Dilated conv.	3x3	8	1x1	128
Dilated conv.	3x3	16	1x1	128
Conv.	3x3	1	1x1	128
Conv.	3x3	1	1x1	128

Retrieval Number F8089088619/2019©BEIESP
DOI: 10.35940/ijeat.F8089.088619
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Deconv.	3x3	1	1x1	64
Conv.	3x3	1	1x1	64
Deconv.	3x3	1	1x1	32
Conv.	3x3	1	1x1	16
Output	3x3	1	1x1	3

The output layer consists of a convolution layer associated with clipping function for output normalization to be in range [0,1]. A detailed architecture of this network is shown in Table 1b. GAN structure is shown in Fig. 3. The presented architecture provides an accurate evaluation for the whole image as well as its local texture's details.

Table 1b. Global Discriminator Network Architecture. The output layer consists of a fully-connected layer with a clipping function that outputs the probability which determines whether the input image belong to real images or an inpainted image.

Type	Kernel	Stride	Outputs
Conv.	5x5	2x2	64
Conv.	5x5	2x2	128
Conv.	5x5	2x2	256
Conv.	5x5	2x2	256
FC	-	-	1

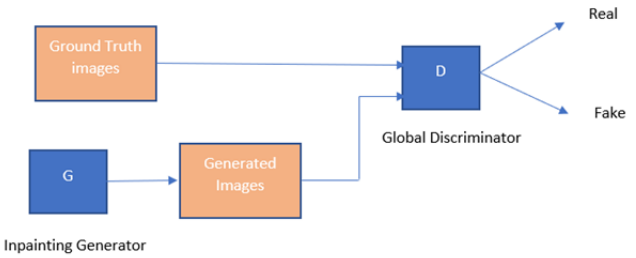

Fig. 3. GAN Architecture. It is trained with $L1_{rec}$ and GAN loss.

Convolution and Mask Modification

For each dilated convolution in the generator inpainting network part, it is joined with a mask modification step. The dilated convolution of every pixel, similarly presented in [28], can be expressed as:

$$x'_{u,v} = \begin{cases} W^T(X \odot M)\dfrac{sum(1)}{sum(M)} + b & if\ sum(M) > 0 \\ 0 & otherwise \end{cases} \quad (1)$$

Where W is weights of the conv. filter, X is the pixel value for the used window, D is the used dilated factor which is =1 for the standard convolution and > 1 for the dilated convolution, M is the used binary mask, i.e. 1 for valid pixels and 0 otherwise, $\odot$ element wise multiplication, $\frac{sum(1)}{sum(M)}$ as a scaling factor for adjusting the unmasked areas' variance and b is used bias.

The modification mask step can be considered as a re-computation of a new mask having fewer missing regions than the previous one. This modification is associated after each convolution layer, but it is conditioned by the ability of the convolution to condition its output on at least one valid input value. The modification mask step is the same update rule used of [28]. If this situation occurs, then the mask is removed for that location. This can be shown by the following:

$$\grave{M}_{u,v} = \begin{cases} 1 & if\ sum(M) > 0 \\ 0 & otherwise \end{cases} \quad (2)$$

This mask step is produced using the same kernel size as the associated convolution layer but with bias set to 0 and weights set to 1. Using an appropriate number of layers with respect to the input image's size, the mask will eventually shrink or fade, i.e. until it is all ones.

Loss Function

For our network to produce coherent and realistic natural images, we used a joint loss function composed of pixel-wise $L1_{rec}$ reconstruction loss and a GAN loss. This joint loss function is optimized in a backpropagation manner using stochastic gradient descent solver, ADAM [33]. ADAM is used for updating network weights in an iterative based in training data. This loss measure was initially used in [1] then adapted by many recent works such as [6-7,30]. A description of each loss function is shown in the following subsections.

Reconstruction Loss: It is used for pixel-wise distance computation between generated image from generator inpainting network and the ground truth image. Even though $L1_{rec}$ forces the network to produce blurry results over complicated textures, this can be explained by the fact that $L1_{rec}$ uses the distribution mean prediction for pixel-wise error minimization.

A solution for this problem is found by adding a GAN loss for a sharper prediction. However, it has a crucial role in directing the network into making sharper predictions in terms of texture color and low frequency details [34]. Also, it plays an essential role as a regularizer for GAN training. $L1_{rec}$ loss is expressed by:

$$L1_{rec} = \frac{1}{N}\sum_{i\sim I}|x(i) - y(i)| \quad (3)$$

Where i is the pixel index and I is the patch. x(i) and y(i) are the pixel values in ground truth patch and generated patch respectively.

GAN Loss: This loss was first introduced by [35]. It is considered as a min-max optimization problem where for each training iteration, the global discriminator is jointly updated with the generator network. Therefore, the optimization is $argmin_G max_D L_{GAN}$ where L_{GAN} is expressed by:

$$L_{GAN}(G,D) = E_{x\sim P(x)}[log D(x)] + E_{y\sim PG(\tilde{x})}[log(1 - D(G(\tilde{x})))] \quad (4)$$

Where: G is the generator network, D global discriminator and $\tilde{x}$ is corrupted image. Therefore, the used joint loss is:

$$L = L1_{rec} + L_{GAN}(G,D) \quad (5)$$

An equal weight for each loss function used with the default hyper-parameter of 1:1. Other loss measures such as perceptual loss, style loss [36] and total variation [37] used in recent work such as [28] are not used in our model as these measures did not result in any remarked any improvement to our results.

After briefly explaining our used method, the algorithm is summarized in Algorithm 1.

IV. EXPERIMENTS AND RESULTS

Random-shaped Mask dataset: Many previous algorithms focused on fixed region inpainting such as square-center regions for inpainting or even randomly scattered regions across the image. This is considered as a limitation from our perspective. To implement a successful algorithm, it must be capable of filling any missing region regardless of its shape, size and location. Therefore, we used the random-shaped mask dataset generated in [28]. All images are resized to 256×256 to cope with the used training set, and we used 10,000 masks for training and validation. Samples of the used random-shaped masks are shown in Fig. 4.

Training Data: We evaluated the performance of our proposed model on two datasets; Places2 [12] and Paris Street View [13] datasets. Places2 is considered as one of the biggest image datasets, it contains 1,803,460 images for training, 36501 for validation and 328501 for testing. Comparison and evaluation are performed using this dataset. Paris Street View consists of 14900 training images and 100 test images, it is only used for general testing of our proposed model.

Algorithm 1. Used Training Algorithm.

```
Algorithm 1 Training Algorithm
1: while iterations < T_train do
2:     For i= 1,2,3,..5
3:     Sample images x from training set with the used batch-size B;
4:     Apply random-shaped masks on x;
5:     Construct courrpted images x̃;
6:     Produce predicted image y;
7:     Calculate joint loss used;
8:     end for
9:     Update G network with joint loss calculated in step 7;
10:    Sample batch images from training data;
11:    Provide random-shapped masks on x;
12: End while
```

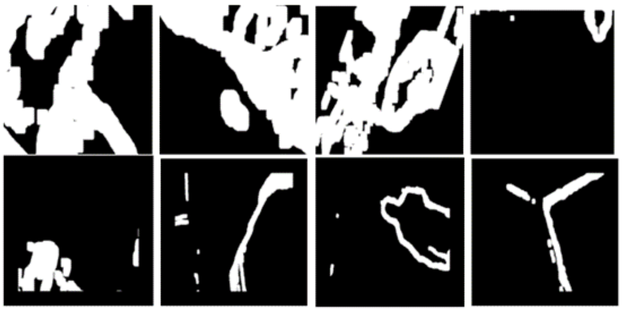

Fig. 4. Random-shaped masks' training samples.

For training and testing, the same original directories for training, validation and testing are used. For Places2, the 256×256 version was used. For Paris Street View, all training and testing images were resized to size 256×256. All our results are a direct output from our model without any fine-tuning or post-processing. Weights are initialized using method in [38] and Adam optimization algorithm [33]. We trained using a one NIVIDIA GPU GTX 1080 (8 GB).

Qualitative Comparison: Results of test images from our model are compared against two different state-of-the-art approaches; Globally and Locally Inpainting [7] (GL) and Generative Inpainting [6] (GI) which are shown in Fig. 5 for Places2 dataset as a reference. Results of Paris Street View dataset is shown in Fig.6.

Quantitative Comparison: The fundamental problem facing image inpainting is the absence of a good quantitative evaluation scheme [6-7,28]. We also followed the evaluation metrics used in recent, previous work. The model performance was reported by mean L1, mean L2 and peak signal-to-noise ratio (PSNR) for Places2 dataset for reference in Table 2. It can be observed that the proposed method gives better measurements for the used error matrices.

Fig. 5. Comparison of our results vs. different methods. It is ordered as Ground Truth, Masked Image, Our Results, GI Results, and GL Results.

User Study: For further analysis and investigations of our model, a human-subjective study is performed. We created a survey to evaluate the naturalness of our results versus the real images using Google Forums.The goal of this survey is to ask users of different ages and backgrounds to evaluate the naturalness of our results when compared to ground truth images. For that purpose, we used 20 randomly picked images from Places2 testing set. We only used the ground truth and generated image from our model, and each user is asked to choose which image appears natural to him/her; First image, second image or both; in case that both images seem natural. Results of our survey are presented in Fig. 7, this also stresses on the superior results from our model. 150 users perform our survey; 26.2% choose the ground truth image as the natural image; 20.8% choose our image as the natural image and 53% choose both images to look natural which proves our very good quality output images.

Limitation of our model: The output model did not show good inpainting results for missing regions that contain a heavily structured object such as human faces or animals as shown in Fig. 8. Also, we did not consider working with higher images resolution.

Fig. 6. Paris Street View results.

Table 2. Quantitative comparison against other methods.

Method	L1 loss	L2 loss	PSNR
GL [7]	8.6%	7.1%	16.90
GI [6]	8.33%	4.7%	16.24
Our Method	**6.45%**	**1.6%**	**19.04**

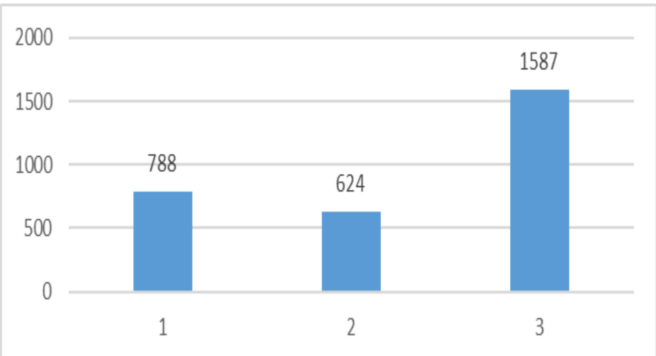

Fig. 7. Results of our survey.

Fig. 8. Failure Cases.

V. CONCLUSIONS

In this paper, a novel-based approach is presented for one of the most challenging task of image inpainting based on dilated convolution and a single global discriminator that works on both local and global structure of images. We showed that using global context discriminator results in trained models that can produce realistic inpainted images. Also, we showed that using dilated convolution can significantly improve the results of the inpainted images with random-shaped missing holes. The proposed model can inpaint any random-shaped missing regions in arbitrary locations in images. We proved our performance through qualitative, quantitative analysis as well as a user study.

REFERENCES

1. Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros, "Context encoders: Feature learning by inpainting", in Proceedings of the IEEE Conference on Computer Vision. 2016, pp. 2536–2544.
2. Yang, Chao and Lu, Xin and Lin, Zhe and Shechtman, Eli and Wang, Oliver and Li Hao, "High-Resolution Image Inpainting Using Multi-Scale Neural Patch Synthesis", in arXiv preprint arXiv:1611.09969, 2017.
3. Nermin M. Salem, Hani M. K. Mahdi and Hazem M. Abbas, "Semantic Image Inpainting using Self-Learning Encoder-Decoder and Adversarial Loss", in 13th International Conference on Computer Engineering and Systems (ICCES), 2018.
4. Ruohan Gao and Kristen Grauman, "On-Demand Learning for Deep Image Restoration", in arXiv preprint arXiv:1612.01380,2017.
5. R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization", in ECCV, 2016.
6. Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu and Thomas S. Huang, "Generative Image Inpainting with Contextual Attention", in arXiv preprint arXiv:1801.07892, 2018.
7. S. Iizuka, E. Simo-Serra and H. Ishikawa, "Globally and locally consistent image completion", in ACM Transactions on Graphics (TOG), 2017.
8. R. Yeh, C. Chen, T. Y. Lim, M. Hasegawa-Johnson and M. N. Do., "Semantic image inpainting with perceptual and contextual losses", in arXiv preprint arXiv:1607.07539, 2016.
9. Y. Li, S. Liu, J. Yang, and M.H. Yang, "Generative face completion", in arXiv preprint arXiv:1704.05838, 2017.
10. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, "Image-to-image translation with conditional adversarial networks", in axXiv preprint arXiv:1611.07004v3, 2018.
11. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., and Fei-Fei, L., "ImageNet Large Scale Visual Recognition Challenge", in International Journal of Computer Vision (IJCV), 2015.
12. B. Zhou, A. Lapedriza, A. Khosla, A. Oliva and A.Torralba, "Places: A 10 million image database for scene recognition", in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.
13. C. Doersch, S. Singh, A. Gupta , J. Sivic and A. Efros, "What makes paris look like paris?", in ACM Transactions on Graphics, 2012.
14. Coloma Ballester, Marcelo Bertalmío, Vicent Caselles, Guillermo Sapiro, and Joan Verder, "Filling-in by joint interpolation of vector fields and gray levels", in IEEE Transactions on Image Processing, 2001.

International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

15. Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles and Coloma Ballester, “Image Inpainting”, in ACM Transactions on Graphics (Proceedings of SIGGRAPH), 2001.
16. M. Bertalmio, L. Vese, G. Sapiro and S. Osher, “Simultaneous structure and texture image inpainting”, in IEEE Transactions on Image Processing, 2003.
17. Leung, Alexei Efros and Thomas, “Texture Synthesis by Non-parametric Sampling”, in International Conference on Computer Vision, 1999.
18. Freeman, Alexei A. Efros and William T., “Image Quilting for Texture Synthesis and Transfer”, in ACM Transactions on Graphics (Proceedings of SIGGRAPH), 2001.
19. Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick, “Graphcut Textures: Image and Video Synthesis Using Graph Cuts”, in ACM Transactions on Graphics (Proceedings of SIGGRAPH), 2003.
20. Connelly Barnes, Eli Shechtman, Adam Finkelstein, Dan B Goldman, “Patchmatch: A randomized correspondence algorithm for structural image editing, in ACM Transactions on Graphics-TOG28, 2009.
21. J. J. Zhao, M. Mathieu, and Y. LeCun, “Energy-based generative adversarial network”, in CoRR, abs/1609.03126, 2016.
22. A. Nguyen, J. Yosinski, Y. Bengio, A. Dosovitskiy, and J. Clune, “Plug and play generative networks: Conditional iterative generation of images in latent space”, in Computer Vision and Pattern Recognition (CVPR), 2017.
23. Jost Tobias Springenberg, “Unsupervised and semi-supervised learning with categorical generative adversarial networks”, in International Conference on Learning Representations (ICLR), 2016.
24. E. Denton, S. Chintala, A. Szlam, and R. Fergus, “Deep generative image models using a laplacian pyramid of adversarial networks”, in arXiv preprint: arXiv:1506.05751v1, 2015.
25. Karen Simonyan, and Andrew Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, in arXiv preprint arXiv:1409.1556, 2015.
26. Wand, C. Li and M., “Combining markov random elds and convolutional neural networks for image synthesis”, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
27. Jonathan Long, Evan Shelhamer, and Trevor Darrell, “Fully convolutional networks for semantic segmentation”, in IEEE Conference on Computer Vision and Pattern Recognition, 2015.
28. Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao and Bryan Catanzaro, “Image Inpainting for Irregular Holes Using Partial Convolutions”, in arXiv preprint arXiv:1804.07723, 2018.
29. Olaf Ronneberger, Philipp Fischer and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation”, in arXiv preprint arXiv:1505.04597v1, 2015.
30. Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang, “Free-Form Image Inpainting with Gated Convolution”, in arXiv perprint arXiv:1806.03589, 2018.
31. Koltun, Fisher Yu and Vladlen, “Multi-Scale Context Aggregation by Dilated Convolutions”, in International Conference on Learning Representations, 2016.
32. Takeru Miyato, Toshiki Kataoka, Masanori Koyama and Yuichi Yoshida, “Spectral normalization for generative adversarial networks”, in arXiv preprint arXiv:1802.05957, 2018.
33. Diederik P. Kingma, and Jimmy Ba, “Adam: A Method for Stochastic Optimization”, in arXiv preprintarXiv:1412.6980, 2014.
34. Unal, Ugur Demir and Gozde, “Patch-Based Image Inpainting with Generative Adversarial Networks”, in arXiv prepint arXiv:1803.07422, 2018.
35. Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville and Yoshua Bengio, “Generative Adversarial Networks”, in arXiv preprint arXiv:1406.2661, 2014.
36. J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution”, in European Conference on Computer Vision, 2016.
37. Shubhangi N. Ghate, Shilpa Achaliya, and Smitha Raveendran, “An Algorithm of Total variation for Image Inpainting”, in International Journal of Computer and Electronics Research, 2012.
38. He, K., Zhang, X., Ren, S., Sun, J., “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”, in Proceedings of the IEEE international conference on computer vision, 2015.