Generative Adversarial Networks for Inpainting Occluded Face Images

Riya Shah, Anjali Gautam[*]*, Satish Kumar Singh*
Information Technology, Computer Vision and Biometrics Lab (CVBL), IIIT Allahabad, Uttar Pradesh, India
riyashah2497@gmail.com, {anjaligautam, sk.singh}@iiita.ac.in   * Corresponding Author

2022 IEEE 6th Conference on Information and Communication Technology (CICT) | 978-1-6654-7312-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/CICT56698.2022.9997943

***Abstract*—Convolutional neural network (CNN) recognizers have made substantial progress in face recognition. Existing recognizers have a lot of power over un-occluded faces, but their performance suffers when it comes to recognizing occluded faces directly. As occlusions cause a lack of visual and recognition signals. The face inpainting task is complicated as it requires generating new pixels for the missing regions of the face image. Generative adversarial networks (GAN) are more suitable for this task when we have to reconstruct visually plausible occlusions in face inpainting. The GAN model is able to generate and inpaint the missing regions of the image. In this paper, we have developed a methodology that makes use of GAN and contextual attention to inpaint images. This image inpainting has applications in the area of face recognition, face animation, and generating synthetic data.**

***Index Terms*—Convolution neural network (CNN), Generative adversarial network (GAN), Image inpainting, Contextual attention.**

I. Introduction

Our visual environment is rich in variety, but it is also structured, and we humans have an extraordinary capacity to decipher this pattern. Take a look at Fig. 1b for example. Even though some parts of the face are missing, the face's content can be easily visualized by us based on the neighbouring pixels, even if we've never seen it. Our minds can easily predict the masked content of the face, even though the faces of humans are rich in diversity and structure. A computer understands only machine language. An algorithm that can impart this intelligence to a computer is required, and it has many application areas such as face recognition, face animation, generating new face images, etc.

Researchers from the past are working on this problem. The algorithm developed by Bertalmio [1] in 2000 based on Partial Differential Equations achieved exemplary results. The method was a huge success, but it could predict the pixels and fill the holes of small sizes only. Many researchers improved upon and came up with better algorithms [2] [3] [4]. With the advancement of technology and with increasing computation power in this era of artificial intelligence, it is important to create deep learning based models that can produce photorealistic images by using large data with less computational power.

Deep Learning based models have produced exemplary outcomes in the task of inpainting the parts of the face that are missing [4]. When it comes to reconstructing missing parts of the face in a visually plausible way, Generative adversarial networks (GAN) based methods [5]–[9] are the suitable choice. Thus, we used GAN-based architecture to do face inpainting. The work in this paper is carried out by taking inspiration from the work reported in paper [10], which is based on the contextual attention mechanism. The proposed methodology is able to fill large holes in the image. Fig. 1 indicates the basic example of image inpainting.

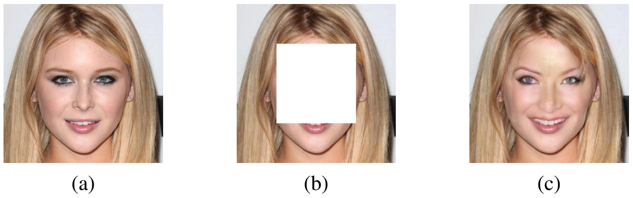
(a) (b) (c)

Fig. 1: Face inpainting example: Real image, b) Masked input image and c) Inpainted image.

II. Related Works

Goodfellow et al. [5] developed GAN architecture and taking inspiration from their work, Pathak et al. [6] proposed Context Encoders (CE) by using conditional GAN. The encoder of CE attempt to map the hidden representations and learn how the pixels of filled and missing regions are related. This information is then used by the decoder to identify pixels and fill in the missing region. Training the model with adversarial loss helped in generating sharper results. Their model received exemplary results as it could generate globally semantic images.

To assure coherency in local image Iizuka et al. [2] enhanced the method reported in [6] by adding one more discriminator. Moreover, they used a dilated-convolution layer to enlarge the receptive field and obtained good results on CelebA dataset with an image size of 256×256 and mask size of 160×160. For refining the reconstructed images, they used Poisson blending. Li et al. [3] developed deep generative completion model which has generator encoder-decoder architecture [11] and two adversarial discriminators for synthesizing missing contents from random noise. They introduced semantic parsing network for enhancing the harmony of the generated contents and existing pixels. The parsing network was primarily an encoder model [12]. The losses then generated were feed-backed to the generator, which helped in

978-1-6654-7312-5/22/$31.00 ©2022 IEEE

improving inpainting results. However, their model failed to bring semantic coherence among inpainted and nearby pixels.

Liu et al. [13] developed a model using Partial Convolutions to inpaint irregular sized and shaped holes. Their model was based on automatic-mask update at each layer so that the partial convolution layer could work only on the non-hole region. Ge et al. [4] tried to address the problem of occluded face recognition and developed Identity-Diversity GAN (ID-GAN) model. In their model, CNN face recognizer was integrated into the GAN model. The recognizer helped the Generator to build faces with a more photo-realistic effect by preserving their identities. Wu et al. [14] used a network based on semantic image inpainting, which is an improvement over the model proposed in [15]. For the generator part, they used the Boundary Equilibrium GAN (BEGAN) and for the discriminator, they used Self-Attention GAN (SAGAN) and replaced the convolution layers with resblocks.

Cheng et al. [16] proposed two-discriminator network based improved GAN model where they overcome over-fitting issue. Their algorithm had the repairing and the discriminator network. The repairing network makes use of simplified Patch-match [17] algorithm to find the closest similar-fit block for filling missing region of the image. The discriminator network has global and local discriminators as in [2]. The algorithm performs well only with regular shaped masks and fails with irregular shaped masks. The other model developed by Peng et al. [18] which was inspired by the hierarchical vector quantized variational auto-encoder (VQ-VAE), and was able to identify structural and textural information in image. Zeng et al. [19] proposed Aggregated COntextual Transformation (AOT) GAN. Their generator consists AOT block, which is a stack of multi layers for context reasoning. A discriminator on the other hand trained by custom mask prediction task for creating fine-grained textures. The AOT blocks use a split-transform-merge technique [19]. The other variant of GAN for image inpainting was given by Liu et al. [9] which had two new modules: (1) the channel and spatially adaptive batch normalization (CSA-BN) module, and (2) the selective latent-space-mapping-based contextual attention (SLSM-CA) layer. CSA-BN used to attenuate the spatial mean and shift in variance in each channel. They also integrated SLSM-CA layer into their model for explicitly capturing long-range correlations.

The existing methods produce blurry, distorted reconstruction results at the boundary region between the hole and the non-hole region, leading to visually unpleasant results. Thus, fail to learn the features from the distant background, which influences the missing pixels in the hole region. To overcome this problem, we have used dilated convolution layers and the Contextual Attention (CA) model. The CA model helps in improving the sharpness of the reconstruction results. It tries to improve the spatial coherence of the reconstructed part of the image.

III. Dataset

The increasing use of deep learning based models in inpainting tasks, masks and data are essential elements for training and assessing the efficacy of the models. In face inpainting, some common datasets that are being used are CelebA, CelebA-HQ, LFW dataset, etc. Here, we have used CelebA-HQ dataset for training and testing.

Karras et al. [20] generated the CelebA-HQ dataset from CelebA [21] dataset, which comprises of 30,000 high-quality images which have dimensions of 1024×1024, 512×512, and 128×128. Initial dimension of images in CelebA ranges from 43×55 to 6732×8984, with varying backdrops, and each is processed using multiple image quality standards, which ensures that the face is in the central part of the image. Some images from the CelebA-HQ dataset are depicted in Fig. 2.

Fig. 2: Some images taken from CelebA-HQ dataset [21].

IV. Methodology

After the emergence of GAN based methods for image inpainting. The proposed methodology also follows GAN architecture with Contextual Attention model [10]. Here, generator is made by stacking two networks. The stage one network is a coarse network with an encoder-decoder architecture. The second stage is a Refinement network, which is explained below. Two discriminators are used to bring semantic coherence in the generated image, namely, local and global discriminators. The network's input is a 256×256 image with a rectangle hole of size 128×128 in center during training.

A. Generator

The generator receives as input pairs of images containing holes filled with white pixels and a binary mask denoting the hole locations; then it outputs the final generated image. Generator is made up of a two-stage coarse to fine network which generates photo-realistic results.

To fade out the missing parts of the input masked image, in stage one, that is, coarse network employs a dilated convolutional encoder-decoder network which is trained with the help of reconstruction loss. Dilated Convolution expands the kernel (input) by putting holes between its elements. The l (dilation factor) option specifies how much the input is expanded. In other words, the kernel skips (l-1) pixels dependent on the value of this option. The coarse prediction is fed into the second network (refinement network), which predicts improved results. The reconstruction loss is used in training coarse network, whereas the reconstruction, along with GAN losses, is used in training a refinement network.

Fig. 3 shows the architecture of GAN with contextual attention [10]. The stage two (refinement) network perceives a more accomplished scene than that of the input image with some missing parts because of which its encoder learns better features than that of the coarse network. The refinement network is made up of two parallel encoders. The bottom

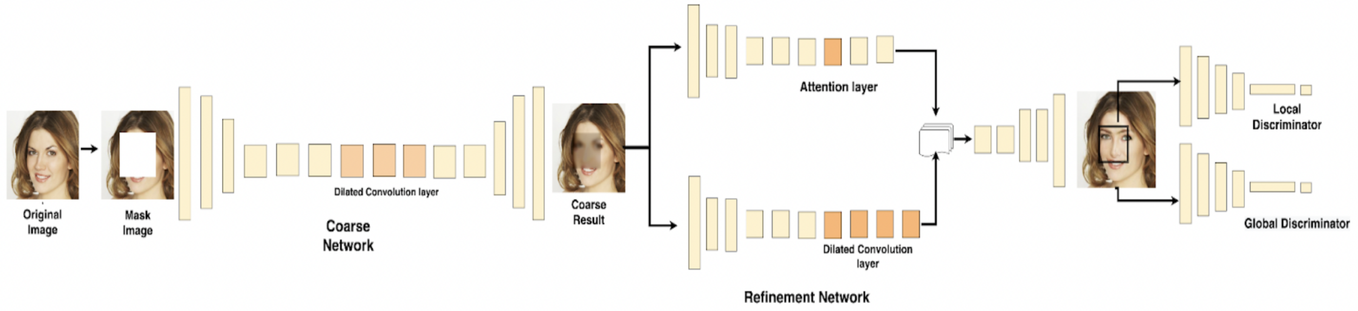

Fig. 3: GAN With Contextual Attention Model [10].

encoder uses layer-by-layer (dilated) convolution to focus on content, whereas the top encoder pays attention to background characteristics of interest. The top encoder is a contextual attention model. To obtain the final output, the output features generated from top and bottom encoders are combined and then given as input to the decoder.

B. Contextual Attention(CA)

CNNs use local convolutional kernels to analyze image features layer by layer, hence they are ineffective at borrowing characteristics from faraway spatial regions. To generate missing patches, the contextual attention layer learns where the information about features is to be borrowed or copied from the patches of background which are known. To begin, we utilize convolution to calculate a matching score between foreground and background patches by means of convolutional filters. After that, we use the softmax to compare and each pixel's attention is rated. Finally, deconvolution is performed on the attention score to rebuild foreground patches with background patches. The contextual attention layer is fully convolutional.

C. Discriminator

With minor reconstruction errors, the training of generator is done to fill the masked region. It doesn't, however, guarantee that the filled area is visually realistic and consistent. We use a discriminator, which acts as a binary classifier to differentiate between images that are real and fake, to promote more photo-realistic effects. The aim of discriminator is that it helps in improving the quality of synthesized output so that unrealistic images do not trick the qualified discriminator.

The model contains two discriminators, namely local and global. The global discriminator focuses on full image and it also maintains coherence. The local discriminator just considers the patch of image inpainted and maintains its coherency with surroundings, thereby ensuring local coherency. Moreover, dilated convolution layer is used to enlarge the receptive field and for refining the image. The architecture of discriminator is similar to [2].

D. Loss Functions

The reconstruction loss is used at the time of coarse network training, whereas Refinement network is trained with reconstruction loss, generator loss, local and global discriminator loss. Reconstruction loss is calculated as spatial discounting loss multiplied by l1-norm of real and fake image. Spatial discounting loss is a distance matrix that holds the values of the distance (l) between the nearest known pixel and the masked pixel. Then it is taken to the power of λ where its value is set to 0.99 for all experimental purposes. Spatial Discounting loss helps in better predicting the pixels near the center of the masked region, which has more ambiguity than those near the boundary of the hole.

Here, Wasserstein GAN (WGAN) [22] is used, where Generator and Discriminator follow the WGAN loss equations. WGAN loss is known for excelling existing losses of GAN for the task of image inpainting work. Its performance is better when paired with l1-reconstruction loss as both employ l1-distance as a metric. The values of the pixels are first clipped so that they fall in the range of -1 and 1 before calculating generator and discriminator loss. This is done as we are interested in the distance between the two images and not the probability of distribution. The model aims at maximizing generator loss and minimizing discriminator loss, thereby following the general trend of min-max loss of GAN architectures. While training, generator loss is the combination of reconstruction loss and its own loss, where reconstruction loss is first multiplied with a constant of 1.2 (based on experiments) and then added to the generator loss. So Generator loss (L_G) is as follows:

$$L_G = 1.2 * L_r + G_{Loss} \tag{1}$$

here, L_r is reconstruction loss and G_{Loss} is negation of mean of the fake image generated by Generator. In **Algorithm 1**, x represents the set of input images and m represents the center square regular mask, $\tilde{x}$ represents the output of Generator. $\tilde{x}$ is also fed as input to Global Discriminator and $\hat{x}$ is an input to the Local Discriminator. t represents the matrix of 0's and 1's, which is synonymous with the mask m. Line 7 is used to make the input of the Local Discriminator. The reconstructed image is processed using the equation mentioned in line 7 so that it masks the section surrounding the inpainted part of the result and then the extraction of the center square region of size 128×128 is done.

Fig. 4 represents the work flow diagram of our methodology.

Algorithm 1 Algorithm for Contextual Attention Model

```
do
    for $j$ in Range 1 to 5 do
        $x$ images are sampled in batches from data to be trained
        $m$ random masks are generated for $x$
        masking of images $y \leftarrow x \cdot m$
        Obtain Results $\tilde{x} \leftarrow y + G(y, m) \cdot (1 - m)$
        $t \sim U[0, 1]$ and $\hat{x} \leftarrow (1\text{-}t)x + t\tilde{x}$
        Using $x, \tilde{x}, \hat{x}$ update both the discriminators
    end for
    $x$ images are sampled in batches from data to be trained
    $m$ random masks are generated for $x$
    Update the Generator using different Loss
while G has not converged
```

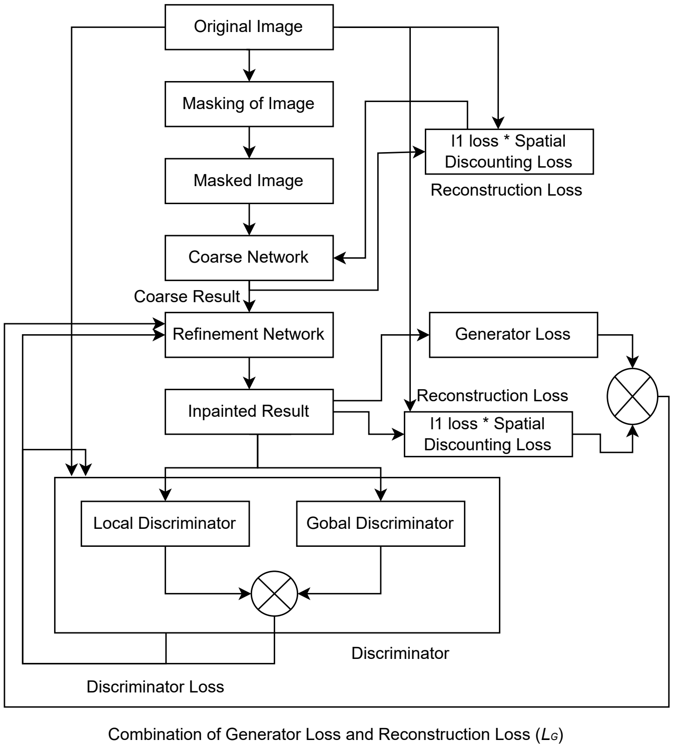

Fig. 4: Block diagram of our methodology

V. Experimental Setup

All the training and testing experiments are done on the Apple MacBook-M1 Chip. It is a 16-core architecture and M1 chip is built to excel at the tasks performed in machine learning, featuring an 8-core GPU, accelerators for machine learning, and a Neural Engine. It has four cores, each of which is designed for running a single task with the aim of maximising performance. It consumes less power as its high-performance silicon core is the quickest in the world. There is a significant improvement in performance, because it has four cores and each of them is multithreaded.

The model is trained on CelebA-HQ dataset. The mask size is 50% of the image size, as if we increase beyond this the model needs a powerful machine, large training time, and would not give satisfactory results. For mask size less than 50% model was giving somewhat satisfactory results as some part of face visible in those sizes. Thus, our aim is to check the performance of model on 50% mask size. Here, the model has trained over 28,000 images with a batch size of 16 and for 10,000 epochs, and testing is done over 2000 images. The overall training time is 140 hours. Exponential Linear Unit (ELU) and Adam are used as an activation function and optimizer respectively. The input to the Local Discriminator is 128 × 128 patch as it concentrates only on the center hole region which is to be inpainted. The input to the global discriminator is 256×256 as it takes the entire image as input.

VI. Results

For quantitative comparison, the two prominent metrics namely Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) have been used.

PSNR [23] : It is a ratio that is used to measure the quality between the original and compressed image. Here, instead of a compressed image, an inpainted image is used.

To calculate PSNR, first the mean-squared error (MSE) of the test image (Y) and the reference image (X) is calculated:

$$MSE(X,Y) = \frac{1}{MN} \Sigma_{i=1}^{N} \Sigma_{j=1}^{M} (X_{ij} - Y_{ij})^2 \tag{2}$$

Here, M and N indicate number of rows and columns. PSNR is calculated using equation (3):

$$PSNR(X,Y) = 10 \cdot log_{10} \left(\frac{R^2}{MSE(X,Y)} \right) \tag{3}$$

SSIM [23] : It is also used to compute the similarity between two images. For inpainting, it will make a comparison between the ground truth image and the image that is inpainted by the model. It is calculated for the given test image (Y) and the reference image (X) using the following formula:

$$SSIM(X,Y) = l(X,Y)^{\alpha} \cdot c(X,Y)^{\beta} \cdot s(X,Y)^{\eta} \tag{4}$$

here $\alpha = \beta = \eta = 1$ by default.

$$l(X,Y) = \frac{2\mu_X \mu_Y + C1}{\mu_X^2 + \mu_y^2 + C1} \tag{5}$$

$$c(X,Y) = \frac{2\sigma_X \sigma_Y + C2}{\sigma_X^2 + \sigma_Y^2 + C2} \tag{6}$$

$$s(X,Y) = \frac{\sigma_{XY} + C3}{\sigma_X \sigma_y + C3} \tag{7}$$

where equation (5), (6) and (7) represents the luminous, contrast and structure comparison functions respectively. This qualitative comparison can be applied to the local and global regions.

Fig. 5 shows the visual results obtained on CelebA-HQ dataset for a central mask of 128 × 128 in an image of size 256 × 256. We have achieved an average PSNR and SSIM of 26.11 and 0.85 respectively over the entire testing dataset.

TABLE I: Comparative analysis of previous works on CelebA-HQ dataset.

Method	**Mask type**	**Mask Size**	**PSNR**	**SSIM**
Proposed Methodology	Central Square Regular	50%	26.11	0.85
Peng et al. [18]	Central Square Gray	50%	24.56	0.867
AOT-GAN [19]	irregular	50%	24.06	0.834
PatchGAN [24]	irregular	50%	24.47	0.849
HM-PatchGAN [19]	irregular	50%	24. 63	0.853
SM-PatchGAN [19]	irregular	50%	24.65	0.852

Table I shows the prominent works carried out on CelebA-HQ dataset. For all the methods mentioned in the table except [18], training was done on 28,000 images and testing was done on 2,000 images, however, for [18] testing was done on 1,000 images.

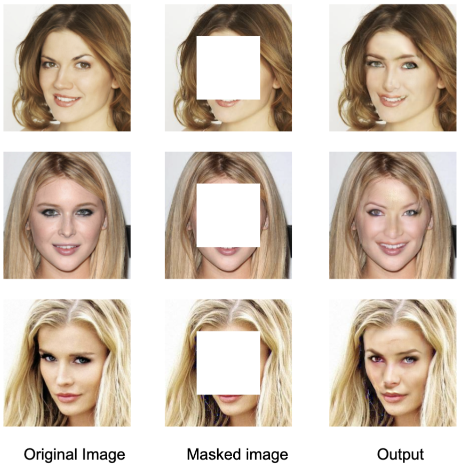

Fig. 5: Generated result with 50% masking

VII. Conclusion

In this work, we attempted to develop an image inpainting methodology based on deep learning by using Contextual Attention. We performed a detailed analysis of the methodology. The reason behind obtaining better PSNR and SSIM values is the Contextual Attention model. The model learns from the available known pixels of the background and then predicts the unknown pixel. In simple terms, it helps in improved feature learning, which is important for inpainting visually plausible results. In the future, this model can be taken forward to improve it in terms of training time as it has huge training time and can be made a part of bigger deep learning projects based on image inpainting, restoring some parts of videos etc.

References

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 417–424.

[2] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, pp. 1–14, 2017.

[3] Y. Li, S. Liu, J. Yang, and M.-H. Yang, "Generative face completion," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3911–3919.

[4] S. Ge, C. Li, S. Zhao, and D. Zeng, "Occluded face recognition in the wild by identity-diversity inpainting," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 10, pp. 3387–3397, 2020.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[6] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2536–2544.

[7] M.-Y. Liu, X. Huang, J. Yu, T.-C. Wang, and A. Mallya, "Generative adversarial networks for image and video synthesis: Algorithms and applications," *Proceedings of the IEEE*, vol. 109, no. 5, pp. 839–862, 2021.

[8] A. Wali, Z. Alamgir, S. Karim, A. Fawaz, M. B. Ali, M. Adan, and M. Mujtaba, "Generative adversarial networks for speech processing: A review," *Computer Speech & Language*, vol. 72, p. 101308, 2022.

[9] J. Liu, M. Gong, Z. Tang, A. K. Qin, H. Li, and F. Jiang, "Deep image inpainting with enhanced normalization and contextual attention," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2022.

[10] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5505–5514.

[11] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." *Journal of machine learning research*, vol. 11, no. 12, 2010.

[12] J. Yang, B. Price, S. Cohen, H. Lee, and M.-H. Yang, "Object contour detection with a fully convolutional encoder-decoder network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 193–202.

[13] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 85–100.

[14] C. Wu, Y. Xian, J. Bai, and Y. Jing, "Semantic image inpainting based on generative adversarial networks," in *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*. IEEE, 2020, pp. 276–280.

[15] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5485–5493.

[16] Y. Chen, H. Zhang, L. Liu, X. Chen, Q. Zhang, K. Yang, R. Xia, and J. Xie, "Research on image inpainting algorithm of improved gan based on two-discriminations networks," *Applied Intelligence*, vol. 51, no. 6, pp. 3460–3474, 2021.

[17] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics*, vol. 28, no. 3, p. 24, 2009.

[18] J. Peng, D. Liu, S. Xu, and H. Li, "Generating diverse structure for image inpainting with hierarchical vq-vae," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 775–10 784.

[19] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Aggregated contextual transformations for high-resolution image inpainting," *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[20] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[21] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[22] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in neural information processing systems*, vol. 30, 2017.

[23] A. Horé and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th International Conference on Pattern Recognition*, 2010, pp. 2366–2369.

[24] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.