R-MNet: A Perceptual Adversarial Network for Image Inpainting

Jireh Jam[1], Connah Kendrick[1], Vincent Drouard[2], Kevin Walker[2], Gee-Sern Hsu[3] and
Moi Hoon Yap[1]

[1] Manchester Metropolitan University, Manchester, UK
[2] Image Metrics Ltd, Manchester, UK
[3] National Taiwan University of Science and Technology, Taipei, Taiwan

Abstract

Facial image inpainting is a problem that is widely studied, and in recent years the introduction of Generative Adversarial Networks, has led to improvements in the field. Unfortunately some issues persists, in particular when blending the missing pixels with the visible ones. We address the problem by proposing a Wasserstein GAN combined with a new reverse mask operator, namely Reverse Masking Network (R-MNet), a perceptual adversarial network for image inpainting. The reverse mask operator transfers the reverse masked image to the end of the encoder-decoder network leaving only valid pixels to be inpainted. Additionally, we propose a new loss function computed in feature space to target only valid pixels combined with adversarial training. These then capture data distributions and generate images similar to those in the training data with achieved realism (realistic and coherent) on the output images. We evaluate our method on publicly available dataset, and compare with state-of-the-art methods. We show that our method is able to generalize to high-resolution inpainting task, and further show more realistic outputs that are plausible to the human visual system when compared with the state-of-the-art methods. https://github.com/Jireh-Jam/R-MNet-Inpainting-keras

1. Introduction

Image restoration is achieved through the process of image inpainting, a technique initially performed by hand to restore images damaged by defects (e.g. cracks, dust, spots, scratches) to maintain image quality. Recently, image inpainting has taken a digital format, and is defined in computer vision as applying sophisticated algorithms that interpolate pixels for disocclusion, object removal and restoration of damaged images. Research in this area of study has been propelled by the increased demand for photo editing in mobile applications (*e.g.* Snapseed, lightroom, pixlr express and flickr), where modifications are done by erasing

unwanted scenes and/or recovering occluded areas. Images contain visible structural and textural information which when distorted can be easily recognized by the human visual system. Maintaining image realism is therefore of utmost importance. Several methods have attempted to maintain image realism. These methods can be classified into two groups: Conventional and deep learning methods. Conventional methods approach image inpainting using texture synthesis techniques via mathematical equations, to obtain image statistics from surrounding pixel similarity for best fitting pixels to fill in missing regions caused by defects. These methods [6, 3, 2, 4] use extended textures from local surrounding of similar pixels to fill in missing regions. Using patches of similar textures by Barnes et al. [2], this technique can synthesize content. However, it lacks high-level semantic details and often generates structures that are non-realistic with repetitive patterns.

In contrast, the second group of approaches (deep learning methods) [26, 13, 39, 38, 22, 23, 37, 40, 16] uses generative neural networks to hallucinate missing content of an image based on encoding the semantic context of the image into feature space for realistic output by a decoder. This is done through convolutions which is an operation that extracts feature maps by evaluating the dot product between a kernel and each location of the input image. The convolutional features are usually propagated channel-wise or through a fully connected layer to a decoder for reconstruction and may sometimes yield images: 1) that are overly smooth (blurry) and 2) with texture artefacts that lack edge preservation. Usually the lack of meaningful semantic content on inpainted images by the state of the art can result from applying the loss function on the entire image, which evaluates the error of the image as a whole, instead of focusing on predicted pixels in masked regions. In this paper, we propose a novel generative neural network, namely R-MNet that predicts missing parts of an image and preserves its realism with complete structural and textural information. Our algorithm takes into account the global semantic structure of the image and predicts fine texture details for

the missing parts that are consistent and realistic to the human visual system.

To enable and enhance the reversed masking mechanism, we propose a reverse-mask loss in feature space that measures the distance between predicted output of the mask regions and the corresponding original pixel areas of the mask on the input image. To achieve this, we use features from VGG-19 pre-trained on ImageNet by Simonyan et al. [31]. We revisit Context-Encoder by Pathak et al. [26] and without bias, we make significant changes that overcome the limitations encountered by existing state of the art. We design an architecture that focuses on the missing pixel values, to extract features and encode them in latent space in an end-to-end fashion. In summary, our main contributions are:

- We propose an end-to-end Reverse Masking Wasserstein GAN image inpainting framework (R-MNet) with improved performance when compared to the state of the art.

- The proposed reverse masking technique can improve the quality of inpainting results by applying the reversed mask on the masked-image as target regions for inpainting.

- A perceptually motivated new combination loss function defined using high level features to target missing pixels to train the novel R-MNet to produce images of high visual quality.

Our approach has achieved high-quality results. The output images when compared with the state of the art have more coherent texture and structures similar to the original images without any post processing.

2. Related Work

Recent work on image inpainting is predominantly focused on deep learning methods due to their ability to capture distribution of high-dimensional data (images). The use of neural network on image inpainting was first approached by Jain et al. [15] as an image denoising task formulated with parameter learning for backpropagation with the noisy image as the learning problem. However, restricted to one colour channel with a single input layer and limited to noise, this model was extended to inpainted images but required substantial computation.

Pathak et al. [26] introduced the use of adversarial training to image inpainting and propose the context-encoder that combines pixel-wise reconstruction loss with adversarial loss by Goodfellow et al. [8] to predict missing pixel values on images. Iizuka et al. [13], improved upon the Pathak et al. [26] and proposed the use of two discriminators; a local discriminator to assess the consistency of predicted

pixels and a global discriminator to assess the coherency of the entire image. Yang et al. [38] introduced a combined optimisation framework, a multi-scale neural synthesis approach with a constraint to preserve global and local texture during pixel-wise prediction of missing regions. Yeh et al. [39] proposed to search closest encoding with context and prior loss combined with adversarial training for the reconstruction of images in latent space. Liu et al. [23] propose partial convolutions with automatic mask updating combined with re-normalised convolutions to target only valid (missing) pixel prediction. Yan et al. [37] added a special shift connection layer that serves as a guidance loss to the U-NET [28] architecture for deep feature rearrangement of sharp structures and fine texture details.

Yu et al. [40] introduced a coarse-to-fine network with contextual attention layer that replaces the Poisson blending post-processing step in [13]. Wang et al. [34] proposed a Laplacian pyramid approach, supported by residual learning [10] that propagates high frequency details to predict missing pixels at different resolutions. Huang et al. [12] proposed an image completion network-based adversarial loss combined with L_1 and Structural Similarity Index Measure (SSIM) [35] to improve on the structural texture and authenticate the reconstructed image. Zeng et al. [42] proposed cross-layer attention and pyramid filling mechanisms to learn high level semantic features from region affinity to fill pixels of missing regions in a pyramid fashion. Li et al. [21] revisited partial convolution [23] and introduced a visual structure reconstruction layer to incorporate the structural information in the reconstructed feature map. Liu et al. [24] proposed a coherent semantic attention layer on a dual-network embedded in the encoder as a refinement mechanism. Ren et al. [27] introduced a two-stage model for structure reconstruction and texture generation with the help of appearance flow in the texture generator to yield image details.

In 2019, Yu et al. [41] proposed the use of gated convolutions in image inpainting that automatically learns soft mask from data. This network [41] combines gated convolutions with an attention layer to automatically learn a dynamic feature mechanism for each channel at each spatial location. Guo et al. [9] proposed the use of partial convolutions in residual network with feature integration determined by dilation step that assigns several residual blocks as a one dilation strategy combined with a step loss for intermediate restoration. Wang et al. [32] used a multi-scale attention module that utilizes background content. Xie et al. [36] introduced bidirectional attention maps that target irregular hole filling within the decoder. Wang et al [33] reused partial convolutions combined with multi-stage attention that computes background and foreground scores from pixel-wise patches. within the decoder to improve quality during the reconstruction of irregular masked regions. However,

these methods have a common practice of using multiple networks as part of the generator or discriminator and expensive post-processing to perform an image inpainting task and do not consider using visible information that targets the missing pixels.

3. Proposed Framework

In this section, we present our R-MNet for solving image inpainting tasks. We employ as base a Wasserstein Generative Adversarial Network (WGAN), using the encoding-decoding architecture of WGAN networks we introduce our new reverse masking operator that enforces the model to target missing pixels thus allowing the network to recover hidden part of an image while keeping the visible one. We also defined a new loss function namely reverse masking loss build around this reverse masking operator.

3.1. Network Architecture

As mentioned previously, R-MNet is build using a GAN as base architecture. GANs have been previously used in image inpainting as they are able to generate missing pixels, unfortunately this often leads to the introduction of blurriness and/or artefact effects. Recent works by Liu et al. [23], Guo et al. [9] and Yu et al [41] try to solve this problem by using partial convolution and gated convolutions. While these two approaches aim to target more efficiently missing pixels we found that they do not fully reduced the aberrations. Our aim through the reverse masking operator is to better target missing region in the image while keeping visible pixels intact.

First, we need to define some generic terminology that will be used through the rest of the paper. We define the source image as $\mathcal{I}$, the mask as M and the reversed masked $M_r = 1 - M$. The masked input image $\mathcal{I}_M$ is obtained as follows:

$$\mathcal{I}_M = \mathcal{I} \odot M \qquad (1)$$

where $\odot$ is the element wise multiplication operator.

Our network architecture is designed to have a generator (G_θ) and a Wasserstein Discriminator (D_θ). Our generator is designed with convolutional and deconvolutional (learnable up-sampling) layers. The convolutional layers encode features in latent space during convolution. These layers are blocks of convolution with filter size of 64 and the kernel size set to 5×5 with a dilation rate of 2 and Leaky-ReLU, α=0.2. We included dilated convolution to widen the receptive field to capture fine details and textural information. The convolutional feature maps obtained in each layer are the input to the next layer after rectification and pooling. We use Maxpooling to reduce variance and computational complexity by extracting important features like edges, and keep only the most present features. We include in our learnable up-sampling layers, reflection padding on a ker-

nel size that is divisible by the stride (K-size=4, stride=2), and bilinear interpolation to resize the image, setting the up-sampling to a high-resolution, and through a tanh function output layer. The goal of setting up the decoder in this way is to ensure that any checker-board artefacts [29] associated with the inpainted regions on the output image are cleaned and consistent with details outside the region. This technique is equivalent to sub-pixel convolution achieved in [30]. We include specifically the WGAN adopted from [1] that uses the Earth-Mover distance, as part of our network to compare generated and real distributions of high-dimensional data. The generator will produce a predicted image $\mathcal{I}_pred = G_\theta(\mathcal{I}_M)$. Using our reversed masked operator we obtain M_r and combined it with $\mathcal{I}_{pred}$ to produce predicted masked area image:

$$\mathcal{I}_{Mpred} = \mathcal{I}_{pred} \odot M_r \qquad (2)$$

The overall architecture is shown in Figure 1. By using this approach, our model predicts only regions with missing pixels which are consistent with surrounding pixels close to border regions of the original image. This results in high-quality reconstructed images that match the natural texture and structure of the original images that are visually plausible with preserved realism.

3.2. Loss Function

3.2.1 Generator Loss Function

We define our generator loss function $\mathcal{L}_G$ to evaluate two aspect of the predicted image: the quality of missing pixels area and, the whole image perceptual quality. Building $\mathcal{L}_G$ around these two metrics will ensure that the generator produces accurate missing pixels that they will blend nicely with the visible pixels. State-of-the-art methods [7, 17, 23, 24, 9] contribute to style transfer and image inpainting have used feature space instead of pixel space to optimize network. Using feature space encourages adversarial training to generate images with similar features, thus achieving more realistic results. Our new combination of loss function is computed based on feature space. We achieve this by utilizing pre-trained weights from the VGG-19 model trained on ImageNet [20]. We extract features from block3-convolution3 and compute our loss function using Mean Square Error (MSE) [26] as our base. Instead of using pixel-wise representations, we use extracted features and compute the squared difference applied to the input and output of our loss model as our perceptual loss ($\mathcal{L}_p$), which is similar to [17], as in equation 3:

$$\mathcal{L}_p = \frac{1}{\kappa} \sum_{i \in \phi} (\phi_i[\mathcal{I}] - \phi_i[\mathcal{I}_{pred}])^2 \qquad (3)$$

where κ is the size of ϕ (output from block3-convolution3 of VGG19), $\phi_i[I]$ is the feature obtained by running the forward pass of VGG19 using I as input and $\phi_i[\mathcal{I}_{pred}]$ is the
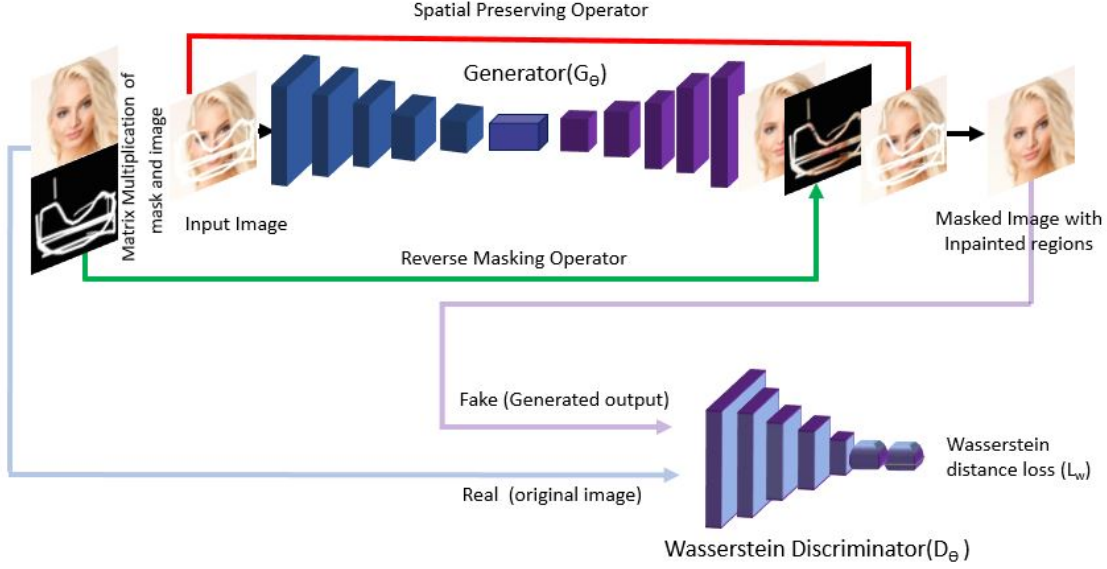
Figure 1: An overview of R-MNet architecture at training showing the spatial preserving operation and reverse-masking mechanism.

feature obtained by running the forward pass on the output of the generator $G_\theta[I_M]$.

We define our reversed mask-loss ($\mathcal{L}_{rm}$) on the same bases as MSE, but targeting only valid features created by the mask region for reconstruction. Our reversed mask loss compares the squared difference for corresponding pixels specific for regions created by the mask on the image and the reconstructed pixels of the masked-image. We use the reversed mask (M_r) and the original image (X_{input}) to obtain $\mathcal{L}_{rm}$, where

$$\mathcal{L}_{rm} = \frac{1}{\kappa} \sum_{i \in \phi} (\phi_i[\mathcal{I}_{Mpred}] - \phi_i[\mathcal{I} \odot M_r])^2 \qquad (4)$$

Finally by linearly combining $\mathcal{L}_p$ and $\mathcal{L}_{rm}$ we obtain the generator loss function:

$$\mathcal{L}_G = (1 - \lambda)\mathcal{L}_p + \lambda\mathcal{L}_{rm}, \qquad (5)$$

where $\lambda \in [0\ 1]$, to allow an optimal evaluation of features by minimising the error on the missing region to match predictions comparable to the ground-truth.

3.2.2 Discriminator Loss Function

Since we train our network with Wasserstein distance loss function ($\mathcal{L}_w$), we define this in equation 6.

$$\mathcal{L}_w = E_{\mathcal{I} \sim P_x}[D_\theta(\mathcal{I})] - E_{\mathcal{I}_{pred} \sim P_z}[D_\theta(\mathcal{I}_{pred})] \qquad (6)$$

Here the first term is the probability of real data distribution and the second term is the generated data distribution.

3.3. Reverse Mask

We discuss the advantages of our approach using reverse mask operator compared to Partial Convolution (PConv) and Gated Convolution (GC), two approaches previously used for image inpainting. All three methods are summarized in Figure 2. The process in partial convolution layers takes in both the image and mask to produce features with a slightly filled mask. Each partial convolutional layer has a mask which if renormalised focuses on valid pixels and an automatic mask update for the next layer. With more partial convolution layers, the mask region gets smaller and smaller, which can disappear in deeper layers and revert all mask values to ones. With gated convolutions, the convolutions automatically learn soft mask from data, extracting features according to mask regions. Each convolution block learns dynamic features for each channel at each spatial location and pass it through different filters. In this process, the output features goes through an activation mechanism (ReLU) while gating values (between zeros and ones) goes through sigmoid function. The gating values show learnable features with semantic segmentation and highlighted mask regions as a sketch in separate channels to generate inpainting results. This network requires a substantial amount of CPU/GPU memory to run the gating scheme. Our proposed reverse mask forces the convolutions to subtract the non-corrupt areas through the reverse masking mechanism ensuring final predictions of the missing regions, with the help of the reverse mask loss, forcing the network via the backward pass to focus on the predictions of the missing regions yielding more plausible outcomes.
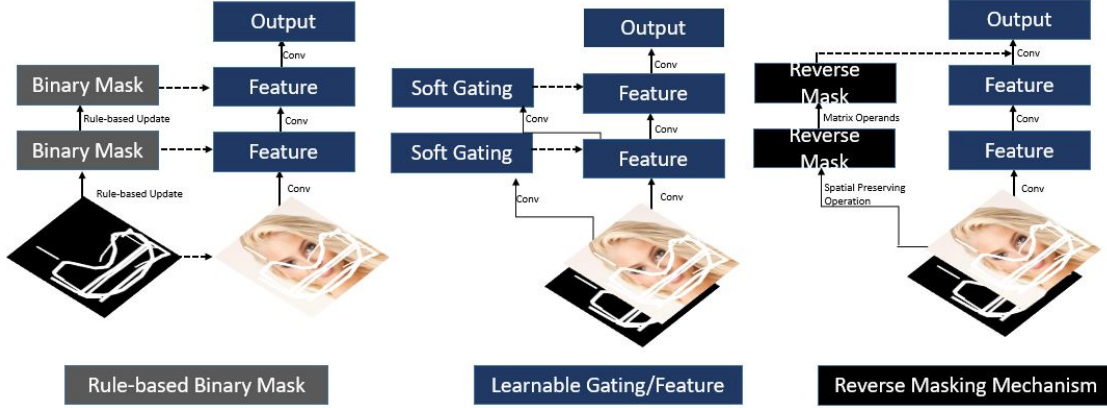
Figure 2: Illustration of partial convolution (left) and gated convolution (middle) and Reverse-masking (right).

Figure 3: Sample images from Quick-Draw Dataset by Iskakov et al. [14]

4. Experiment

In this section, we introduce datasets used in this work and present our implementation. We evaluate our approach qualitatively and quantitatively, demonstrating how effective our method is in performing image inpainting without any post-processing mechanism.

4.1. Datasets

We use three publicly available datasets to evaluate our method: CelebA-HQ [18], Places2 [43] and the Paris Street View [5]. For masking, we use the Quick Draw mask dataset [14] shown on Figure 3, which contains 50,000 train and 10,000 test sets designed based on 50 million human drawn strokes in vector format combined to form irregular shapes (patterns) [14] of size 512×512 pixels. During training and testing, we randomly select the masks from various sets and resize each to 256×256. We used the CelebA-HQ curated from the CelebA dataset [25] by Karras et al. [18]. From the 30,000 high quality images of 1024×1024, 512×512 and 128×128 resolutions, we followed the state-of-the-art procedures [23] and split our dataset into 27,000 training and 3,000 testing set. The Places2 [43] dataset contains more than 10 million images with more than 400 unique scene categories with 5,000 to 30,000 train images. We split the training and testing set according to the state-of-the-art [26] and trained our network to understand scene category. The Paris Street View has 14,900 training images and 100 validation images. We use the same testing set as described in the state-of-the-art [26] for our experiment.

4.2. Implementation

We use the Keras library with TensorFlow backend to implement our model. Our choice of datasets matches the state of the art [23, 24, 26, 24] with similar experimental settings. We resize all images and masks using OpenCV library interpolation function INTER_AREA to $256 \times 256 \times 3$ and $256 \times 256 \times 1$ respectively. We use the Adam optimizer[19] with learning rate of 10^{-4}, $\beta = 0.9$ for G_θ and 10^{-12}, $\beta = 0.9$ for D_θ. We train our model with a batch size of 5 on NVIDIA Quadro P6000 GPU machine, on Places2 and Paris Street View. We use NVIDIA GeForce GTX 1080 Ti Dual GPU machine on CelebA-HQ dataset high-resolution images. It takes 0.193 seconds to predict missing pixels of any size created by binary mask on an image, and 7 days to train 100 epochs of 27,000 high-resolution images.

5. Results

To evaluate the performance of the proposed method, we compare R-MNet with three other methods on the same settings for image size, irregular holes and datasets. Our experiments include

- **CE**: Context-Encoder by Pathak et al. [26]

- **PConv**: Partial Convolutions by Liu et al. [23]

- **GC**: Free-Form image inpainting with Gated Convolutions by Yu et al. [41]

- **R-MNet-0.1**: R-MNet using ℓ_{rm} when $\lambda = $ **0.1**, our proposed method.

- **R-MNet-0.4**: R-MNet using ℓ_{rm} when $\lambda = $ **0.4**, our proposed method.

Figure 4: Visual comparison of the inpainted results by **CE**, **PConv**, **GC** and **R-MNet** on CelebA-HQ [23] where Quick Draw dataset [14] is used as masking method using mask hole-to-image ratios [0.01,0.6].

5.1. Qualitative Comparison

We carried out experiments based on similar implementations by Liu et al. [23], Pathak et al. [26] and pre-trained model for the state of the art [41] and compared our results.

For Places2 dataset, we randomly select 10,000 training samples to match [26] and use the same number of test samples to evaluate our model. We show our results on Figure 4. For CelebA-HQ, we downloaded pre-trained models for the state of the art [41] and compared our results. Based on visual comparison, our model shows realistic and coherent output images. Observing from Figure 4, other models fail to yield images with structural and textural content as the images are either blurry or fail due to the image-to-hole ratio increase with arbitrary mask.

5.2. Quantitative Comparison

To statistically understand the inpainting performance, we quantify our model with the state of the art and compared our results, based on four classic metrics: Frechet Inception Distance (FID) by Heusel et al. [11], Mean Absolute Error (MAE), Peak Signal to Noise Ratio (PSNR), and SSIM [35]. The FID measures the quality of reconstructed images by calculating the distance between feature vectors of ground-truth image and reconstructed images. The other metrics (MAE, PSNR, SSIM) evaluate at pixel and perceptual levels respectively. The results in Table 1 are evaluated based on masks with various test hole-to-image area ratios ranging from [0.01,0.6], on a test set of 3000 images from
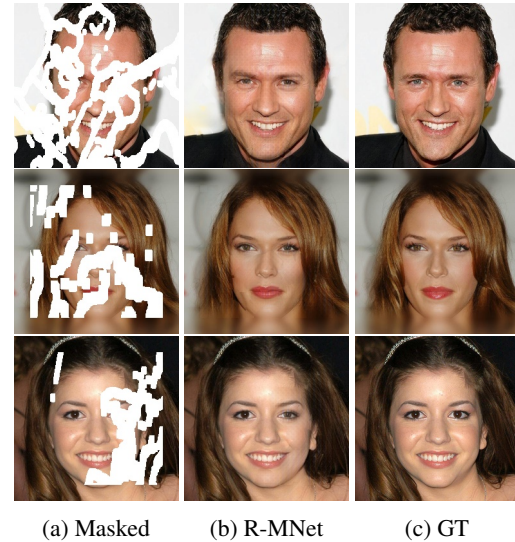

Figure 5: Results of image inpainting using R-MNet-0.4 on CelebA-HQ Dataset [23] with Nvidia Mask dataset [23] used as masks, where images in column (a) are the masked-image generated using the Nvidia Mask dataset [23]; images in column (b) are the results of inpainting by our proposed method; and images in column (c) are the ground-truth.

the CelebA-HQ.

A lower FID score indicates that the reconstructed images are close to the ground-truth. A similar judgement

(a) Masked	(b) R-MNet	(c) GT

Figure 6: Results of image inpainting using R-MNet-0.4 on Places2 [43] and Paris Street View [5], where images in column (a) are the masked-image generated using the Quick-Draw dataset [14]; images in column (b) are the results of inpainting by our proposed method; and images in column (c) are the ground-truth.

Table 1: Results from CelebA-HQ test dataset, where Quick Draw dataset by Iskakov et al. [14] is used as masking method with mask hole-to-image ratios range between [0.01,0.6]. † Lower is better. ⊎ Higher is better.

Inpainting Method	FID †	MAE †	PSNR ⊎	SSIM ⊎
R-MNet-0.1	26.95	33.40	38.46	0.88
Pathak et al. [26]	29.96	123.54	32.61	0.69
Liu et al. [23]	15.86	98.01	33.03	0.81
Yu et al. [41]	4.29	43.10	39.69	0.92
R-MNet-0.4	**3.09**	**31.91**	**40.40**	**0.94**

Table 2: The inpainting results of R-MNet-0.4 on Paris Street View and Places2, where Quick Draw dataset by Iskakov et al.[14] is used as masking method with mask hole-to-image ratios range between [0.01,0.6]. † Lower is better. ⊎ Higher is better.

Dataset	FID †	MAE †	PSNR ⊎	SSIM ⊎
Paris Street View	17.64	33.81	39.55	0.91
Places2	4.47	27.77	39.66	0.93

quantifies the MAE, though it measures the magnitude in pixel error between the ground-truth and reconstructed images. For PSNR and SSIM, higher values indicate good quality images closer to the ground-truth image. Looking

at the results in Table 1, our model achieves better performances than other state-of-the-art methods.

To test the effectiveness of our model, we conduct a test using the Nvidia Mask dataset [23], and show our results on the CelebA-HQ dataset in Figure 5. These masks are of different categories with hole-to-image area ratios. Note that these masks were not used during training of R-MNet. We used it for testing only to demonstrate our model superiority and robustness across mask. We carry out further experiments on Paris Street View and Places2 to generalize our model. Masks of the same sizes [0.01,0.6] are randomly selected during testing. Results can be seen on Table 2 and Figure 6, which shows our model is able to generalize to various inpainting tasks and not just face inpainting.

5.3. Ablation Study

We investigate the effectiveness of reversed mask loss, and conduct experiments at different λ and compare its performance using hole-to-image area ratios between [0.01, 0.6]. We use $\lambda=$ **0, 0.1, 0.3, 0.4, 0.5** on reversed mask loss for different experiments with the same settings. The results are shown in Table 3. When $\lambda=$**0** the model has no access to the reversed mask to compute the loss function, we realise that the mask residue is left on the image as shown on Figure 7. Based on the output images, although the spatial information of the image is preserved, the convolutional inpainted regions need assistance to minimise the loss between the mask and the reverse mask loss during prediction. We start by giving a small value to $\lambda=$**0.1**, we notice that the results get better but we obtain poor performance visually. That is, the pixels in the mask region did not blend properly with the surrounding pixels, leaving the image with inconsistencies in texture. We experiment further with $\lambda=$ **0.3, 0.4, 0.5** and carried out subjective evaluations using FID, MAE, PSNR and SSIM. With $\lambda=$ **0.4**, we obtained the best results with no further improvement by increasing the value of λ. The mask as input to the CNN allows the network to learn the size of the corrupted region. The bigger the mask, the longer it takes to achieve perceptual similarity. This is because the region grows bigger and the network takes longer due to a smaller proportion of the loss covering the entire image to ensure the inpainted region is semantically consistent with the rest of the image.

6. Discussion

The ability to generalize good performance on machine learning algorithms based on end-to-end mapping of real data distribution to unseen data, is vital to the learning outcome by various models. In image inpainting, algorithms based on generative networks, predict missing regions following a real data distribution from a large dataset. Typical approaches predict these hidden regions by applying an encoding-decoding process to the image, where the missing

Table 3: Results from Paris Street View and Places2 using Quick Draw dataset by Iskakov et al. [14] as masking method with mask hole-to-image ratios [0.01,0.6]. † Lower is better. ⊎ Higher is better.

$\mathcal{L}_{rm}$ weight	FID †	MAE †	PSNR ⊎	SSIM ⊎
λ=0.1, R-MNet-0.1	26.95	33.40	38.46	0.88
λ=0.3, R-MNet-0.3	4.14	31.57	40.20	0.93
λ=0.4, **R-MNet-0.4**	**3.09**	**31.91**	**40.40**	**0.94**
λ=0.5, R-MNet-0.5	4.14	31.0	40.0	0.93

(a) Masked (b) $\lambda = 0$ (c) $\lambda = 0.1$ (d) $\lambda = 0.4$ (e) GT

Figure 7: Visual results on ablation study where (a) is the input masked image (b) results of the R-MNet-Base model without ℓ_{rm}. As loss on this model we used ℓ_2 and binary-cross-entropy as loss (c) R-MNet with ℓ_{rm} loss at with weight application of 0.1 (d) R-MNet with full $\ell_{VP}(\phi)$ with λ=0.4 on Quick-Draw [14] as masking method.

regions are defined usually by a binary mask. The encoding process will produce a high dimensional feature representation of the image where the missing information has been recovered, while the decoding process will generate the original image, i.e. the input image without missing information.

Generally the learning procedure of the model parameters is performed by solving a loss function minimization. Often, the model parameters are learned using a forward-backward process. In the forward pass the loss function calculates an error between latent distribution of real and generated data. The loss is then back-propagated into the model to update the parameters weight (backward pass).

The ability of our model to identify the missing regions of the input image is essentially assisted by our reverse mask loss, that will force the network to primarily focus on the prediction of the missing regions. The reverse mask loss combined with the perceptual loss and the preserved spatial information within the network, will ensure accurate prediction of missing regions while keeping the general structure of the image and high resolution details.

A practice of existing inpainting GAN is the requirement to apply the mask on an image to obtain a composite image

(masked-image), which in turn share the same pixel level. Additionally, when using irregular masks, these regions to be inpainted can be seen by the algorithm as a square bounding box that contains both visible and missing pixels, which can cause the GAN to generate images that can contain structural and textural artefacts.

To overcome this issue, we modify the usual approach by having two inputs to our model, during the training, the image and its associated binary mask. This allow us to have access to the mask, at the end of our encoding/decoding network, through the spatial preserving operation and gives us the ability to compute the following, that will be used during the loss computation:

- A reversed mask applied to the output image.

- Use a spatial preserving operation to get the original masked-image.

- Use matrix operands to add the reversed mask image back on the masked-image.

By using this method of internal masking and restoring, our network can inpaint only the required features while maintaining the original image structure and texture with high level of details. Our network shows better achievement, when compared to state-of-the-art methods, numerically and visually, where the output image are visually closer to the original image than other approaches.

Globally we demonstrate through our approach that combining global (perceptual) and specific (reverse mask) loss we can achieve better performances, thus overcoming the limitation of having a model trained only using global loss.

7. Conclusion

In this paper, we propose a novel approach using Reverse Masking combined with Wasserstein GAN to perform image inpainting task on various binary mask shapes and sizes. Our model targets missing pixels and reconstructs an image with structural and textural consistency. We demonstrate that our model can perform accurate image inpainting of masked regions on high resolution images while preserving image details. Through our experimental results, we have shown that training our model alongside performing reverse matrix operands of the mask is beneficial to image inpainting. We also show that our model when compared with the state-of-the-art methods can obtain competitive results.

Acknowledgement

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*.

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics*, 28(3):24, 2009.

[3] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *ACM Conference on Computer graphics and interactive techniques*, 2000.

[4] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing*, 13(9):1200–1212, 2004.

[5] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012.

[6] Alexei A Efros and Thomas K Leung. Texture synthesis by non-parametric sampling. In *International Conference on Computer Vision*, 1999.

[7] Leon Gatys, Alexander Ecker, and Matthias Bethge. A neural algorithm of artistic style. *Journal of Vision*, 16(12):326–326, 2016.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[9] Zongyu Guo, Zhibo Chen, Tao Yu, Jiale Chen, and Sen Liu. Progressive image inpainting with full-resolution residual network. In *ACM International Conference on Multimedia*, 2019.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition*, 2016.

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017.

[12] Ying Huang, Maorui Wang, Ying Qian, Shuohao Lin, and Xiaohan Yang. Image completion based on gans with a new loss function. *Journal of Physics: Conference Series*, 1229(1):012030, 2019.

[13] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. *ACM Transactions on Graphics*, 36(4):107, 2017.

[14] Karim Iskakov. Semi-parametric image inpainting. *arXiv preprint arXiv:1807.02855*, 2018.

[15] Viren Jain, Joseph F Murray, Fabian Roth, Srinivas Turaga, Valentin Zhigulin, Kevin L Briggman, Moritz N Helmstaedter, Winfried Denk, and H Sebastian Seung. Supervised learning of image restoration with convolutional networks. In *IEEE International Conference on Computer Vision*, 2007.

[16] Jireh Jam, Connah Kendrick, Vincent Drouard, Kevin Walker, Gee-Sern Hsu, and Moi Hoon Yap. Symmetric skip connection wasserstein gan for high-resolution facial image inpainting. *arXiv preprint arXiv:2001.03725*, 2020.

[17] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, 2016.

[18] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[19] Diederik P Kingma and Jimmy Lei Ba. Adam: A method for stochastic gradient descent. In *International Conference on Learning Representations*, 2015.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[21] Jingyuan Li, Fengxiang He, Lefei Zhang, Bo Du, and Dacheng Tao. Progressive reconstruction of visual structure for image inpainting. In *IEEE International Conference on Computer Vision*, 2019.

[22] Yijun Li, Sifei Liu, Jimei Yang, and Ming-Hsuan Yang. Generative face completion. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[23] 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 *European Conference on Computer Vision*, 2018.

[24] Hongyu Liu, Bin Jiang, Yi Xiao, and Chao Yang. Coherent semantic attention for image inpainting. In *IEEE International Conference on Computer Vision*, 2019.

[25] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision*, 2015.

[26] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[27] Yurui Ren, Xiaoming Yu, Ruonan Zhang, Thomas H Li, Shan Liu, and Ge Li. Structureflow: Image inpainting via structure-aware appearance flow. In *IEEE International Conference on Computer Vision*, 2019.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015.

[29] Adrian Rosebrock. *Deep Learning for Computer Vision with Python*. PyImageSearch.com, 2.1.0 edition, 2019.

[30] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE conference on computer vision and pattern recognition*, 2016.

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[32] Ning Wang, Jingyuan Li, Lefei Zhang, and Bo Du. Musical: Multi-scale image contextual attention learning for inpainting. In *IJCAI*, pages 3748–3754, 2019.

[33] Ning Wang, Sihan Ma, Jingyuan Li, Yipeng Zhang, and Lefei Zhang. Multistage attention network for image inpainting. *Pattern Recognition*, page 107448, 2020.

[34] Qiang Wang, Huijie Fan, Gan Sun, Yang Cong, and Yandong Tang. Laplacian pyramid adversarial network for face completion. *Pattern Recognition*, 88:493–505, 2019.

[35] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[36] Chaohao Xie, Shaohui Liu, Chao Li, Ming-Ming Cheng, Wangmeng Zuo, Xiao Liu, Shilei Wen, and Errui Ding. Image inpainting with learnable bidirectional attention maps. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8858–8867, 2019.

[37] Zhaoyi Yan, Xiaoming Li, Mu Li, Wangmeng Zuo, and Shiguang Shan. Shift-net: Image inpainting via deep feature rearrangement. In *European Conference on Computer Vision*, 2018.

[38] Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[39] Raymond A Yeh, Chen Chen, Teck-Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with deep generative models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[40] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *IEEE conference on computer vision and pattern recognition*, 2018.

[41] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. In *IEEE International Conference on Computer Vision*, 2019.

[42] Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Learning pyramid-context encoder network for high-quality image inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[43] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2018.