CIAFill: Lightweight and Fast Image Inpainting with Channel Independent Attention

1st Chung-Il Kim
Artificial Intelligence Research Center
Korea Electronics Technology Institute
Gyeonggi-do, Korea
cilkim1@keti.re.kr

2nd Saim Shin
Artificial Intelligence Research Center
Korea Electronics Technology Institute
Gyeonggi-do, Korea
sishin@keti.re.kr

3rd Han-Mu Park
Artificial Intelligence Research Center
Korea Electronics Technology Institute
Gyeonggi-do, Korea
hanmu@keti.re.kr

***Abstract*—Image inpainting is a classic technique in computer vision research. The quality of image inpainting has improved significantly since the advent of convolutional neural networks. However, this approach generally results in blurry and semantically inconsistent reconstruction because of operating valid and invalid pixels with equal weight. The gated convolution computing feature attention is proposed to resolve this issue but this attention mechanism was less efficient and computationally expensive. This study proposed CIAFill that alleviates this problem using channel independent attention. This mechanism applied channel attention to each channel for activating valid channels and reduced the computational cost to the dimension of the channel. The proposed architecture included a channel attention generator and a channel attention projection PatchGAN that utilize the channel independent attention mechanism. This study proved that CIAFill could successfully train the inpainting model with 1/1600 smaller gating parameters than the earlier gated convolution-based study. CIAFill achieved comparable performance to other feature attention-based approaches in the experiments on CelebA-HQ and Places2 datasets.**

***Index Terms*—inpainting, attention module, generative adversarial net**

I. Introduction

An image inpainting task involves reconstructing occluded or blank regions to make images plausible by referring to information in surrounding regions. This is a popular task in the field of computer vision and image processing [1]–[3], because corruptions by noise and occlusions frequently occur in real-world [4]–[7].

The performances of image inpainting have been significantly improved with the advent of deep learning technologies [8], so most recent image inpainting techniques are based on deep convolutional neural networks (CNN) [9]–[12]. CNN-based networks trained by massive data can reconstruct highly structured regions including complex semantics—faces, hands, and buildings—because the learning paradigm of CNN involves analyzing the pixel-wise data distribution from training data [13]–[15]. However, CNN-based inpaintings commonly generate implausible results, such as blurry texture, apparent color discrepancy, and abnormal edges around erased regions [16]. These errors occur because the CNN filters indiscriminate to both valid and invalid regions [17].

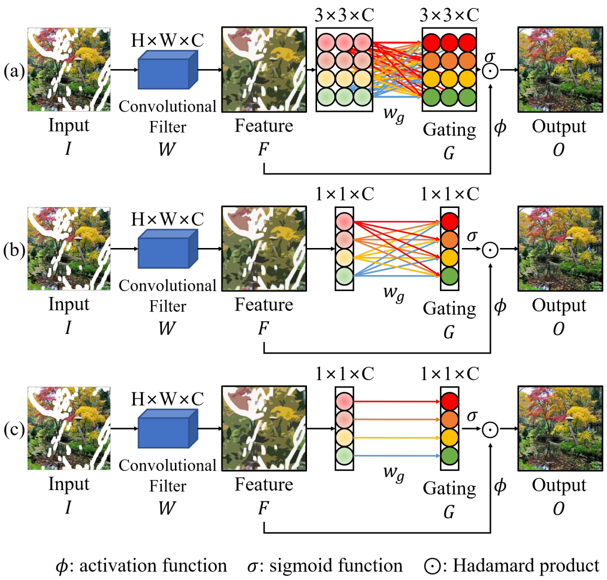

Fig. 1. Architecture of three attention mechanims. (a): feature attention, (b): channel attention, (c): channel independent attention.

The gated convolution (GC) in *DeepFill v2* adopts feature attention in each core block to address this problem and filters out pixels interfering with reconstruction [16]. GC-based architectures [16], [18]–[20] can generate more detailed results than earlier CNN-based approaches in irregular masks [9]–[11], but the performance improvement of feature attention has been negligible compared to the effect of other vision tasks such as image recognition [21]–[23].

The channel attention can improve its performance better than that of the feature attention for high-speed computation and lightweight [24], [25], as shown in Fig. 1(b) and (a), respectively. Although this technique used fewer parameters than the earlier feature attention technique, its performance improved significantly. However, channel attention for such inpainting is still inefficient because two things are not considered: 1) the destruction of direct correspondence between channels and weights owing to changes in the channel dimen-

This work was supported by Institute of Information and communications Technology Planning and evaluation (IITP) grant funded by the Korea government (MSIT) (2021-0-00537, Visual common sense through self-supervised learning for restoration of invisible parts in images).

978-1-6654-5818-4/22/$31.00 ©2022 IEEE

sion and 2) computational amount owing to the total coupling between channels [20], [22], [25].

To address these issues, this paper proposes channel independent attention (CIA) in Fig. 1(c). The proposed CIA focuses on the channels of the current features, activates valid channels, and reduces the parameters required for computing. This paper also proposes a CIAFill architecture that includes a channel attention generator (CAG) and a channel attention projection PatchGAN (CAPP). CAG is a generator that adopts CIA blocks as core modules and requires fewer parameters to adopt the gating concept than the earlier approaches [16], [20]. CAPP is an attention-guided discriminator that also adopts CIA blocks as core modules to boost the quality of reconstructed results.

II. Related work

A. Convolution structure for inpainting

[9] presented deep learning-based inpainting utilizing a CNN with a generative adversarial network (GAN) [26]. Following this study, the use of CNN with GAN has become mainstream in inpainting research [10], [11], [16], [17]. However, because of the limitation of semantic understanding, CNN-based methods commonly yield blurry reconstruction results in complex scenes.

[17] investigated the reason for the blurry results from the CNN-based inpainting models. This study revealed that convolutional filters were spatially shared for all input pixels or features, leading to a blurry reconstruction. This implied that the invalid pixels or features in the region, such as holes, propagated the meaningless information to surrounding regions and generated blurry results. To address this problem, [17] proposed a partial convolution (PC) mechanism that considers only valid pixels for reconstruction by masking invalid pixels and renormalizing. However, this model was stacked by PC, layer to layer, and invalid pixels gradually converted to valid pixels through training; nevertheless, these valid pixels were neglected.

[16] suggested GC utilizing feature attention. The GC provided information for each layer from its previous layer using the feature-wise product. Because of its effectiveness in extracting valid features from its previous layer and the capability to utilize user sketch input to guide the results, GC has been widely used in recent inpainting [18]–[20].

[20] suggested three types of light weight gated convolution (LWGC) - depth-separable LWGC (LWGC^{ds}), pixel-wise LWGC (LWGC^{pw}), and single-channel LWGC (LWGC^{sc}) - for lightweight inpainting and synthesizing high-resolution images. These models reduced the number of parameters but did not outperform GC.

B. Channel attention mechanism

Channel attention has been proposed in image recognition tasks to alleviate the high computation of feature attention [22], [24], [25], [27]. The squeeze-excitation module improved the recognition performance compared to general convolution by compressing and re-activating the channel [24]. The gather-excite module was proposed for a lightweight approach and better context exploitation in CNNs using strided depth-wise convolution [23], [28]. The bottleneck attention module (BAM) divided feature attention into channel attention and spatial attention, and then computed them in parallel [27]. The convolutional bottleneck attention module improved the performance over the BAM by replacing parallel operations with serial operations [22]. The channel attention module efficiently improved the image recognition performance with fewer channel interactions and a similar channel size [25].

C. Generators inside GAN for inpainting

DeepFill v1 [11] introduced two-stage inpainting models with a dilated convolution structure. Strided convolution [29] was used to reduce the resolution of the image by one-quarter for preventing the loss of pixel information and then added five extended layers to increase the receptive field of this structure. However, this module is slow and heavy load because it utilizes two stages and is calculated with high resolution.

U-net was proposed for inpainting structure [17], [30], but these structures utilized a relatively large number of learnable parameters than dilated convolution-base structure.

D. Discriminators inside GAN for inpainting

The most commonly used discriminator in inpainting is PatchGAN [31]. Each dimension of the PatchGAN output receives only the patch region in the images and determines the region as real or fake. This model allows the generator to produce realistic images in image-to-image translation [31], [32], but commonly tends to undergo unstable training [33].

SN-PatchGAN [16] provided more stable training than the earlier work [11] by widening the receptive fields and adopting spectral normalization [33]. This discriminator eliminated the one-channel convolution in PatchGAN and adjusted the adversarial loss on each neuron of the output, leading to fast and stable training. However, this discriminator exhibited poor performance in inpainting largely masked images.

Boundless' discriminator [34] was a modified version of the conditional projection discriminator [35]. This discriminator replaced the classification label input with a pre-trained ImageNet model [36]. Because features from Inception v3 [37] contain more information from the conditioning vector, this structure improved the discriminator in photo-realistic and seamless synthesis. However, this model lost spatial conditions in the discriminating process.

III. Approaches

Figure 2 demonstrates the overall architecture proposed in this study. CIA is an effective structure for fast and correct inpainting. To improve the attention-based GAN structure for the inpainting task, the channel attention generator and channel attention projection PatchGAN are introduced for the generator and discriminator in GAN, respectively.

The contributions of the proposed architecture are as follows. First, this model can achieve higher performance without increasing the number of channels, compared to the earlier

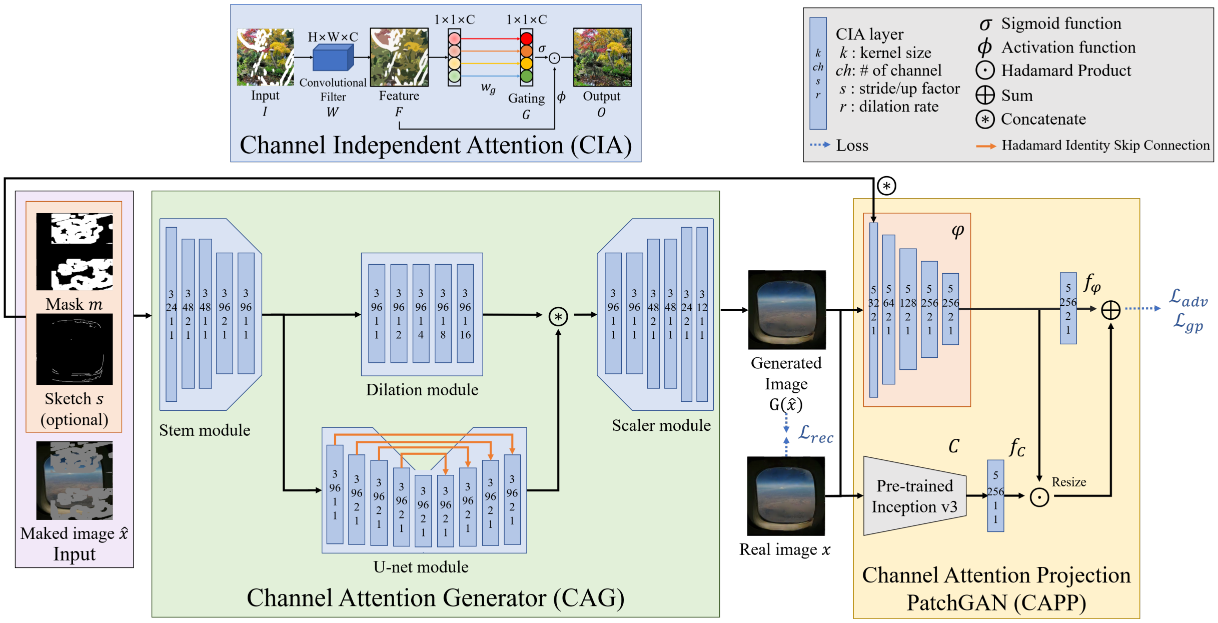

Fig. 2. Architecture of CIA, CAG, and CAPP for image inpainting.

attention-based models [16], [20]. This architecture introduces a serial operation mechanism for high-performance image inpainting. Earlier attention mechanisms for inpainting models such as GC and LWGC applied parallel operations. However, in the modeling process of repeated channel expansions and reductions, the semantic correspondences between channels and their weights with different channel sizes can be doubly destructive in parallel architectures, compared to the proposed serial mechanisms [25]. Next, the channel attentions in this architecture is calculated independently for each channel. This method can expect an effect similar to that of increasing the number of channel parameters and reduces the parameter computation by the number of channels compared to a typical channel attention [23]. Table I shows the number of training parameters required by each convolution mechanism used in inpainting.

TABLE I
THE NUMBER OF PARAMETERS NEEDED TO COMPUTE GATING FOR EACH MECHANISM.

Mechanism	Parameter calculation	k_h,k_w=3 C,C'=32
GC	$k_h \times k_w \times C \times C'$	9216
LWGC ^{ds}	$k_h \times k_w \times C + C \times C'$	1312
LWGC^{pw}	$C \times C'$	1024
LWGC^{sc}	$k_h \times k_w \times C \times 1$	288
CIA (Ours)	C	**32**

A. Channel Independent Attention

The proposed CIA introduces an improved channel attention mechanism for image inpainting. Figure 2 demonstrates the CIA architecture. w_g in this figure shows that each channel of gating G is calculated independently of the channel of feature F with the same index. The CIA formulation is as follows. Assume that input I is C-channel, each pixel located at y, x in C'-channel feature $F_{y,x}$ is computed using Equation (1).

$$F_{y,x} = \sum_{i=-k_h'}^{k_h'} \sum_{j=-k_w'}^{k_w'} W_{k_h'+i,k_w'+j} \cdot I_{y+i,x+j} \tag{1}$$

Where x and y represents x-axis and y-axis of output map respectively, k_h and k_w denote the kernel size. $k_h' = \frac{k_h-1}{2}$, $k_w' = \frac{k_w-1}{2}$, $W \in \mathbb{R}^{k_h \times k_w \times C \times C'}$, and $O_{y,x} \in \mathbb{R}^{C'}$ are inputs and outputs. This mechanism generates G by the given F for reflecting the characteristics of the features. Assume that $G_{y,x}$ is the pixel located at (y, x) in G. The $G_{y,x}$ is expressed as in Equation (2).

$$G_{y,x} = w_g \odot F_{y,x} \tag{2}$$

Where $w_g \in R^{C\prime}$ is a learnable parameter, $\odot$ means Hadamard product [38]. The pixel of output at (y, x), $O_(y, x)$ is computed using Equation (3). σ represents a sigmoid function [39], ϕ represents an activation function. The exponential linear unit is selected in this paper [40].

$$O = \phi(F_{y,x}) \odot \sigma(G_{y,x}) \tag{3}$$

Compared to earlier studies, the proposed CIA is different from spatial attention in vision understanding tasks. Because the input data in the image inpainting task already contain temporal validity for each pixel with masked images, spatial attention focused on the location patterns is unnecessary. In the case of LWGC^{sc} using only spatial attention, poor performance was recorded compared to feature or channel attention-based models [20].

B. Channel Attention Generator

For CAG module in Figure 2, let x, $\widehat{x}$, m, and s represent samples from the original data, erased data, mask, and sketch (optional), respectively. The generator G takes the $\widehat{x}$, m, and s and outputs the generated image $G(\widehat{x})$. The proposed CAG includes four modules: 1) Stem module that reduces the resolution by 1/4 each, 2) dilation module, 3) U-net module, and 4) Scaler module that up-scales the image resolution back to the original.

The stem module has the same parameters as the layers from the first to the fifth of DeepFill v2. Features extracted by this module become the input to the U-Net module and dilation module.

The proposed model mixes a dilated module designed to prevent pixel information loss [10] and a U-Net module that can increase the performance of the model by stacking several layers relative to the former structure [17], [41]. When using the U-Net module, instead of a skip connection that concatenates channels, the Hadamard identity skip connection (HISC) is applied, which can increase inpainting performance and reduce network parameters by replacing valid pixels of the decoder with those of the encoder for each pixel [42]. To avoid breaking the direct correspondence between channels and their weights, both modules consist of the proposed CIA with the same number of channels.

Finally, the scaler module receives the outputs of the dilation and U-Net modules and outputs the image that matches the original resolution.

TABLE II
THE TOTAL NUMBER OF LEARNABLE PARAMETERS FOR EACH INPAINTING MODEL.

Structure	Model	# of learnable parameter
U-net	SC-FEGAN DFNet	42.1M 32.9M
Dilated convolution	EdgeConnect DeepFill v2 HiFill	12.1M 4.1M 2.7M
Both	**CAG (Ours)**	**1.8M**

Table II presents the parameter number of inpainting models according to the model structure, which proves that the proposed CAG acquires the fewest number of parameters. The proposed CAG architecture can achieve high performance with a reduced number of model parameters.

C. CAPP: Channel Attention Projection patchGAN for discriminator

In Figure 2, CAPP represents the proposed discriminator. This discriminator considers the $\widehat{x}$ as fake data with m and s as conditions, and the x as real data with the same m and s as conditions. CAPP is motivated by three representative discriminators. Boundless [34]' discriminator prevents performance degradation in restoring where the erased area is large. An attention-guided discriminator [32] focuses on missing regions. An SN-PatchGAN [16] provides stable learning with global and local features.

To take advantage of these three discriminators, CAPP consists of SN-PatchGAN (ϕ and f_ϕ) as the baseline, pre-trained Inception v3 (C) for extracting perceptual feature of an image, and a convolutional layer f_C matching the dimension of the output by C and ϕ to project its feature as the condition. Formally, our discriminator D is expressed by Equation (4).

$$\begin{aligned} D([\widehat{x}, x, m, s]) &= f_\phi([\widehat{x}, m, s], [x, m, s]) \\ &\quad + f_C(C(x)) \odot \phi([\widehat{x}, m, s], [x, m, s]) \end{aligned} \tag{4}$$

The SN-patchGAN is expressed by Equation (5).

$$D([\widehat{x}, x, m, s]) = f_\phi([\widehat{x}, m, s], [x, m, s]) \tag{5}$$

The Boundless' discriminator is expressed by Equation (6).

$$\begin{aligned} D([\widehat{x}, x, m, s]) &= f_\phi([\widehat{x}, m, s], [x, m, s]) \\ &\quad + \langle f_C(C(x)), \phi([\widehat{x}, m, s], [x, m, s]) \rangle \end{aligned} \tag{6}$$

Where $\langle ., . \rangle$ denotes an inner product. In Equation (4), $f_\phi(\phi([\widehat{x}, m, s], [x, m, s]))$ is from the SN-PatchGAN in Equation (5). $C(x) \odot \phi([[\widehat{x}, m, s], [x, m, s]])$ is from Boundless's discriminator, differs in three respects. First, the inner product is changed to the Hadamard product to preserve the authenticity of each neuron. Second, the features extracted from ϕ can contain sketch information. Although the inpainting task without sketch information was performed in this study, the proposed CAPP model can also receive sketch information as an optional input for user-intended inpaintings. Finally, the Boundless discriminator extracted the features of Inception v3' from the output layer, but the proposed discriminator extracted features from the layer before pooling [37]. This change indicates that this model reflects the derived information of each pixel inferred from the model.

All convolutional layers in this architecture are changed to CIAs for dynamic feature selection. Compared to the earlier attention-guided discriminator with a constant threshold-based attention map [32], the CAPP effectively focuses on the erased parts with CIA layers, and it is possible to extract optimal features for the masked regions.

D. Loss function

The adversarial loss function for this model is adjusted from the loss of SN-patchGAN to reflect the output of each neuron [16]. The model is trained with a mixture of three losses: reconstruction loss L_{rec}, adversarial loss L_{adv} [26], and gradient penalty loss L_{gp} [43] as shown in Equation (7), where $\lambda_{rec} = 100$, $\lambda_{adv} = 1$, and $\lambda_{gp} = 10$ [18], [34].

$$L_{total} = \lambda_{rec} L_{rec} + \lambda_{adv} L_{adv} + \lambda_{gp} L_{gp} \tag{7}$$

The pixel-wise L1 loss is selected as L_{rec}. The hinge loss [44] is applied to L_{adv} [30], [45]. Hinge loss consists

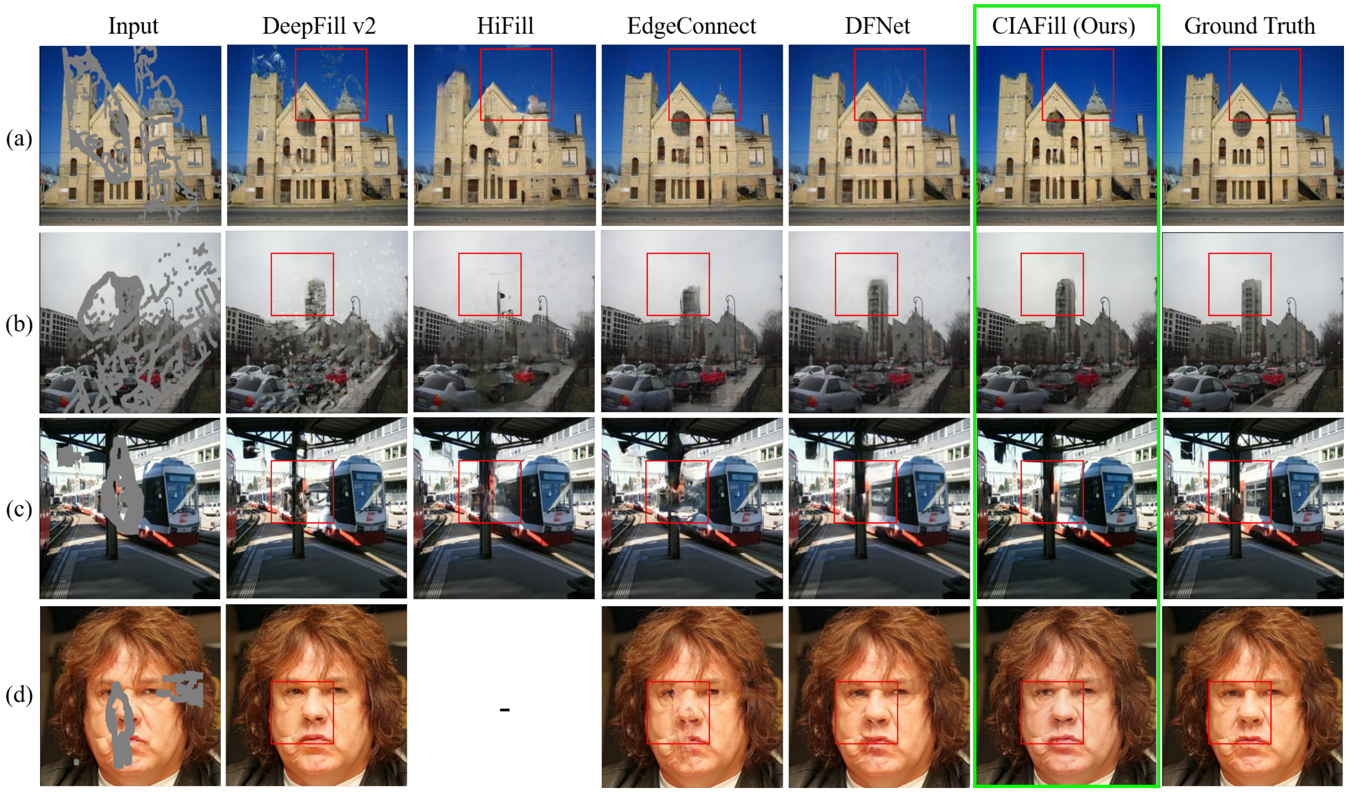

Fig. 3. Examples of a inpainting results using Places2 and CelebA-HQ by each model.

of L_G for training the generator and L_D for training the discriminator. L_G is derived using Equation (8).

$$L_G = -\mathbb{E}_{z\sim\mathbb{P}_Z}[D(G(z))] \tag{8}$$

$\mathbb{E}_{\bullet\sim\mathbb{P}_{\blacksquare}}$ represents the expectation of variables $\bullet$ with probability distribution function $\mathbb{P}_{\blacksquare}$. z represents a set of $\widehat{x}$, m, and s. Z is the probability distribution of z.

The discriminator is trained using Equation (9). $data$ represents the probability distribution of x, $Relu$ is a rectified linear unit [46].

$$\begin{aligned} L_D = \mathbb{E}_{x\sim\mathbb{P}_{data}}&[Relu(1-D(x))] \\ &+ \mathbb{E}_{z\sim\mathbb{P}_Z}[Relu(1+D(G(z)))] \end{aligned} \tag{9}$$

The gradient penalty loss is represented by Equation (10) for high-quality inpainting performances.

$$L_{GP} = \mathbb{E}_{u\sim P_u}[(\|\nabla_u D(u)\|_2 - 1)] \tag{10}$$

In Equation (10), U represents the probability distribution function of u, which is a uniformly sampled data point along the straight line between the discriminator inputs from x and $\widehat{x}$. In this study, the generator and the discriminator used Adam [47] optimizer for training and set the learning rate to 1e-5 and 1e-4 until convergence, respectively.

IV. Experiments

A. Experimental setting

TensorFlow [48] 1.15, CUDA 10 [49], and cudnn 7.4 were used for this experiment. Two computers were used - Intel(R) Xeon(R) Silver 4114 as CPU and Intel(R) Xeon(R) W-2145 as CPU including NVIDIA TITAN RTX as GPU and 64GB of RAM.

Two datasets were used in this experiment: Places2 [50] and CelebA-HQ [51]. Places2 includes 18 million scene photographs with scene categories and is cropped to 256 pixels in both width and height for the experiments. CelebA-HQ is a dataset of face images in which 30,000 high-resolution images are resized to 512 pixels in both width and height. Free-form masks or an irregular mask dataset provided by [17] are used to create irregular holes. The Canny edge algorithm [52] was applied to the datasets to generate the sketch dataset (optional). However, for the sake of fairness, the sketch was not used in the comparative experiments. In all tables in this chapter, the bold fonts and underline indicate the best and the second performance in the same column, respectively. L1 error, Structural SIMilarity (SSIM) [53], and Fréchet Inception Distance (FID) [54] were used as the evaluation metrics to measure how much it is restored like the original, how structurally it is similar to the original, and how similar it is to the original data distribution, respectively.

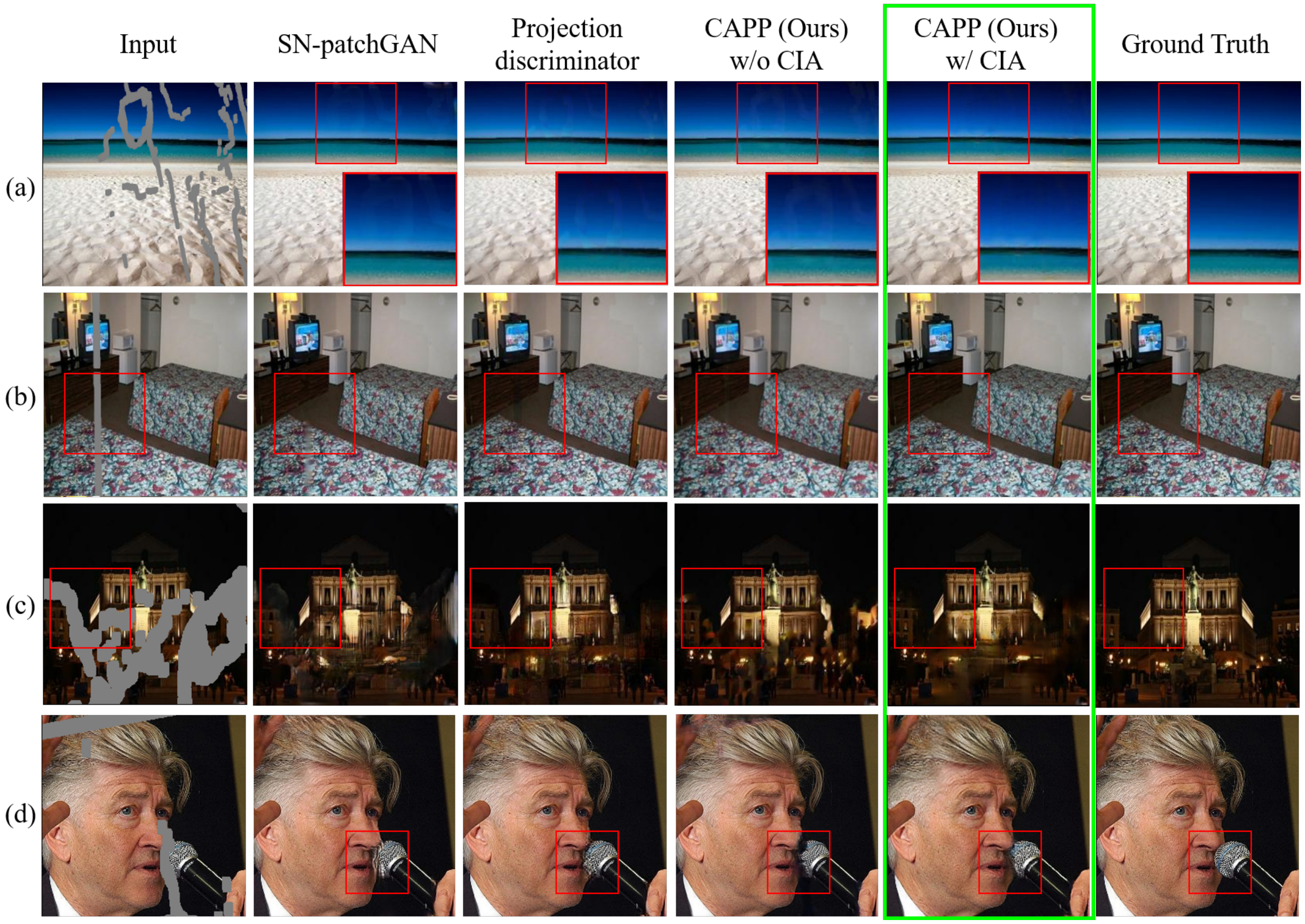

Fig. 4. Examples of inpainting results using Places2 and CelebA-HQ by each discriminator.

B. *Qualitative and quantitative results*

Four models - DeepFill v2 [16], HiFill [20], EdgeConnect [45], and DFNet [30] - were set up to compare the performances with the proposed architecture. The quantitative and qualitative results of each comparative model were implemented using the pre-trained models published by the authors.

Figure 3 illustrates certain inpainting results of the five models. In the case of (a), the image could not be restored by DeepFill v2 without utilizing the NVIDIA irregular mask dataset proposed for general-purpose inpainting. As shown in (b), the proposed CIAFill successfully restored the shape of a building unlike DeepFill v2 or HiFill because CIAFill concentrates only on the pixel index without spatial information during attention. (c) indicated that only two models, DFNet and CIAFill utilizing U-net, consistently inpainted the black pillars because U-net could consider the full context of the image with stacking strided convolution. In the case of (d), edge connect, DFNet makes lips restoration unnatural. Overall, CIAFill architecture visually outperformed the other models.

Table III presents the average inference time per image, L1 error, SSIM, and FID of five models in the Places2 datasets and four models in the CelebA-HQ datasets. Our model performed the best in all four metrics except for the L1 error in the Places2 testset. In CelebA-HQ, the proposed model achieved the best performance in all four metrics.

TABLE III
PERFORMANCES OF INPAINTING RESULTS BY EACH MODELS.

Places2 Model	Time (ms)	L1 (%)	SSIM	FID
DeepFill v2	51	8.94	0.885	8.61
HiFill	43	8.07	0.884	8.54
EdgeConnect	78	**6.98**	0.902	7.97
DFNet	85	7.11	0.905	7.45
CIAFill (Ours)	**39**	7.00	**0.909**	**7.22**
CelebA-HQ Model	Time (ms)	L1 (%)	SSIM	FID
DeepFill v2	69	4.20	0.921	5.87
EdgeConnect	110	4.57	0.910	6.55
DFNet	126	4.11	0.927	6.16
CIAFill (Ours)	**41**	**4.02**	**0.936**	**5.41**

C. *Comparisons of discriminator*

Figure 4 shows examples of generated images when trained by SN-PatchGAN, projection discriminator, CAPP without CIA, and CAPP with CIA. As shown in (a), it was restored seamlessly because CAPP with CIA using the attention module concentrated only on the deleted part. In the case of (b), the complex texture of the quilt could be reproduced because the features of the original image were used for projection during training. However, the projection discriminator that did not consider local features could not erase the mask marks in the narrowly erased area. As confirmed by (c), considering both local features and attention was essential to prevent unintentional spots because the region of interest can be identified. In the case of (d), mike and the celeb lip were distorted in the outputs of SN-PatchGAN and projection discriminator. There, the proposed discriminator with CIA outperformed other discriminators in this experiment.

TABLE IV
EVALUATION PERFORMANCES BY EACH DISCRIMINATOR.

Places2 Discriminator	L1 (%)	SSIM	FID
SN-patchGAN	8.81	0.866	8.82
projection discriminator	7.18	0.893	7.27
CAPP (Ours) w/o CIA	7.25	0.892	7.84
CAPP (Ours) w/ CIA	**7.00**	**0.909**	**7.22**
CelebA-HQ Discriminator	L1 (%)	SSIM	FID
SN-patchGAN	4.22	0.927	5.61
projection discriminator	4.19	0.916	5.47
CAPP (Ours) w/o CIA	4.15	0.933	5.70
CAPP (Ours) w/ CIA	**4.02**	**0.936**	**5.41**

Table IV reports the performances of the proposed discriminator. CAPP with CIA performed the best on all three metrics in both datasets. In this experiment, whereas the Places2 dataset included a variety of foregrounds, the CelebA-HQ dataset included front-facing human faces. Therefore, CelebA-HQ shared more similar features than Places2. In this case, the correct judgment about attention can be a more important contribution to performance improvement than in other cases. Therefore, CAPP without CIA recorded lower performance than projection discriminator in the Places2 dataset. In the CelebA-HQ dataset, CAPP without CIA performed better than the projection discriminator.

D. Comparisons of attention-based mechanisms

In these experiments, DeepFill v2 was the baseline model, and GC, LWGC, and the proposed CIA changed the generator's convolution mechanism. Five perspectives were applied for the analysis measures of results: L1 error, SSIM, FID, the total number of gatings' learnable parameters, and the time it takes to generate an image of 256×256 in the Places2 dataset.

Figure 5 depicts certain samples for qualitative comparison. The results from the LWGC indicated a noticeable problem with regions that were blurred or erased. Because GC and LWGC utilize spatial information for attention, the channel features were reduced, leaving mask-shaped marks when restored.

As shown in Table V, the proposed CIA outperformed the compared mechanisms in all evaluation measures. In this study, it was proved that the proposed CIA is the most efficient gating method in terms of computational amount and speed than earlier methods.

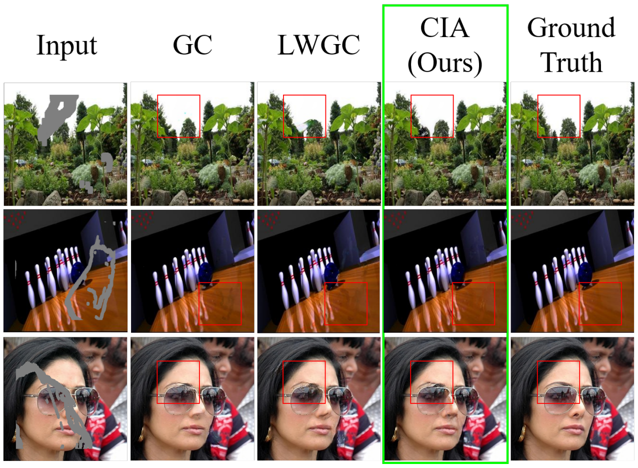

Fig. 5. The inpainting results using Places2 and CelebA-HQ by each attention mechanisms.

TABLE V
PERFORMANCE COMPARISONS OF DEEPFILL V2 BASED MODELS USING PLACES2.

Mechanism	# of gatings' parameter	Time (ms)	L1 (%)	SSIM	FID
GC	1,793,928	51	8.87	0.893	7.65
LWGC	110,564	38	8.82	0.904	7.56
CIA (Ours)	**1,119**	**33**	**8.69**	**0.911**	**7.49**

V. CONCLUSION

This paper introduces the CIAFill architecture for fast and lightweight image inpainting. The CIAFill included CAG and CAPP utilizing the proposed CIA as a core mechanism based on independent channel attention. The experiments in this study proved that the proposed mechanisms performed better than existing models in terms of performance, speed, and model size. The strengths of the proposed methods would contribute to real-time inpainting or inpainting in mobile environments. The CIAFill still requires improvements. It maintains its performance only by using both CAG, which uses both the dilation module and U-Net module, and CAPP, which combines attention-guided discriminator, SN-PatchGAN, and projection discriminator. Therefore, we intend to improve the channel attention mechanism applicable to the general model in our future work.

REFERENCES

[1] A. Criminisi, P. Perez, and K. Toyama, "Object removal by exemplar-based inpainting," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, vol. 2. IEEE, 2003, pp. II–II.

[2] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE transactions on image processing*, vol. 10, no. 8, pp. 1200–1211, 2001.

[3] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image melding: Combining inconsistent images using patch-based synthesis," *ACM Transactions on graphics (TOG)*, vol. 31, no. 4, pp. 1–10, 2012.

[4] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf, "Image completion using planar structure guidance," *ACM Transactions on graphics (TOG)*, vol. 33, no. 4, pp. 1–10, 2014.

[5] D. Liu, X. Sun, F. Wu, S. Li, and Y.-Q. Zhang, "Image compression with edge-based inpainting," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 10, pp. 1273–1287, 2007.

[6] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Deep video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5792–5801.

[7] W. Wang, Q. Huang, S. You, C. Yang, and U. Neumann, "Shape inpainting using 3d generative adversarial network and recurrent convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2298–2306.

[8] S. Esedoglu and J. Shen, "Digital inpainting based on the mumford-shah-euler image model," 2001.

[9] 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.

[10] 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.

[11] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505–5514.

[12] Y. Wang, X. Tao, X. Qi, X. Shen, and J. Jia, "Image inpainting via generative multi-column convolutional neural networks," *arXiv preprint arXiv:1810.08771*, 2018.

[13] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE transactions on neural networks*, vol. 8, no. 1, pp. 98–113, 1997.

[14] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 922–928.

[15] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 833–841.

[16] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4471–4480.

[17] 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.

[18] Y. Jo and J. Park, "Sc-fegan: face editing generative adversarial network with user's sketch and color," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1745–1753.

[19] Y.-G. Shin, M.-C. Sagong, Y.-J. Yeo, S.-W. Kim, and S.-J. Ko, "Pepsi++: Fast and lightweight network for image inpainting," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 252–265, 2020.

[20] Z. Yi, Q. Tang, S. Azizi, D. Jang, and Z. Xu, "Contextual residual aggregation for ultra high-resolution image inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7508–7517.

[21] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3156–3164.

[22] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[23] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-excite: Exploiting feature context in convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 31, pp. 9401–9411, 2018.

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[25] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: efficient channel attention for deep convolutional neural networks, 2020 ieee," in *CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE*, 2020.

[26] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.

[27] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "Bam: Bottleneck attention module," *arXiv preprint arXiv:1807.06514*, 2018.

[28] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[29] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[30] X. Hong, P. Xiong, R. Ji, and H. Fan, "Deep fusion network for image completion," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2033–2042.

[31] 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.

[32] Y. A. Mejjati, C. Richardt, J. Tompkin, D. Cosker, and K. I. Kim, "Unsupervised attention-guided image to image translation," *arXiv preprint arXiv:1806.02311*, 2018.

[33] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.

[34] P. Teterwak, A. Sarna, D. Krishnan, A. Maschinot, D. Belanger, C. Liu, and W. T. Freeman, "Boundless: Generative adversarial networks for image extension," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 10 521–10 530.

[35] T. Miyato and M. Koyama, "cgans with projection discriminator," *arXiv preprint arXiv:1802.05637*, 2018.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[38] R. A. Horn, "The hadamard product," 1990.

[39] F. Rosenblatt, "Principles of neurodynamics. perceptrons and the theory of brain mechanisms," Cornell Aeronautical Lab Inc Buffalo NY, Tech. Rep., 1961.

[40] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[41] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[42] Y. C. E. H. Chung-Il Kim, Jehyeok Rew, "Ufc-net with fully-connected layers and hadamard identity skip connection for image inpainting," *Computers, Materials & Continua*, vol. 68, no. 3, pp. 3447–3463, 2021. [Online]. Available: http://www.techscience.com/cmc/v68n3/42518

[43] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of wasserstein gans."

[44] J. H. Lim and J. C. Ye, "Geometric gan," *arXiv preprint arXiv:1705.02894*, 2017.

[45] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, "Edgeconnect: Structure guided image inpainting using edge prediction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[46] K. Fukushima, "Visual feature extraction by a multilayered network of analog threshold elements," *IEEE Transactions on Systems Science and Cybernetics*, vol. 5, no. 4, pp. 322–333, 1969.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015.

[48] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: A system for large-

scale machine learning," in *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 2016, pp. 265–283.

[49] NVIDIA, P. Vingelmann, and F. H. Fitzek, "Cuda, release: 10.2.89," 2020. [Online]. Available: https://developer.nvidia.com/cuda-toolkit

[50] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1452–1464, 2017.

[51] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3730–3738.

[52] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[53] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[54] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *arXiv preprint arXiv:1706.08500*, 2017.