FMGDN: Flexible Multi-Grained Dilation Network Empowered Multimedia Image Inpainting for Electronic Consumer

Xin Zhang, Bernd Hamann, Dongjing Wang, *Member, IEEE*, Hongbo Wang, Yueyun Wang, Yuyu Yin, *Member, IEEE*, and Honghao Gao, *Senior Member, IEEE*

***Abstract*—Various consumer electronics have become indispensable for our daily lives, and they rely heavily on visual content, such as image, to deliver immersive experiences to users. However, consumer electronic devices may encounter issues like image corruption, noise interference, or object occlusion, and images in consumer electronics may be subjected to various imperfections, which can be addressed by image inpainting techniques to enhance the visual experience. In this work, we propose a Flexible Multi-Grained Dilation Network (FMGDN) to capture multi-grained information and complete hole regions with semantically and visually plausible contents. Specifically, we design a Multi-Grained Residual (MGR) block with hybrid parallel branches to extract hierarchical features. The branches with different dilation rates use co-prime principle to avoid gridding artifacts. The multi-grained features from different branches are fused with adaptive weights to enforce deeper semantic context. Further, a Channel Adaptive Shuffling (CAS) block is designed to effectively decode high-level features back to image space, shuffling the inner channels with learning-capable filters and adaptive channel weights. In particular, benefitting from its concise and flexible architecture, FMGDN can be easily configured to adapt to various kinds of consumer electronics. Experiments on three real-world datasets demonstrate the effectiveness of FMGDN, outperforming state-of-the-art methods.**

***Index Terms*—Image inpainting, Multi-grained, Hybrid dilation, Adaptive fusion, Consumer electronics.**

I. INTRODUCTION

CONSUMER electronics, such as personal computer, smartphones, and televisions, have become indispensable for our daily lives. These devices rely heavily on visual content to deliver immersive experiences. However, images in consumer electronics can often be subjected to various imperfections, such as missing or damaged portions, which can degrade the overall visual quality and user experience. Therefore, many image processing techniques, such as image inpainting [1]–[4], image denoising [5], and image enhancement [6], can be widely applied in consumer electronics to improve the visual quality, including object remover (shawdow removal [7], scratches removal, occlusion removal, reflection removal [8]), restoration, and completion. In particular, image inpainting can remove unwanted text, logos, or watermarks from images or videos, or enhance image quality by filling in missing details and improving the overall visual appearance. Besides, consumer electronic devices may encounter issues like image corruption, noise interference, or object occlusion, which can be addressed by image inpainting techniques with visually appealing and clean images. In particular, millions of consumer electronics users benefit from the integrated image inpainting technologies. However, when it comes to large and irregular hole regions, image inpainting is still a very challenging task involving semantic understanding of the image scenario.

Early image inpainting approaches, such as diffusion-based methods [1] and exemplar-based methods [9], propagate surrounding structures to the hole region or adaptively aggregate non-local similar exemplar patches. These methods are not suitable for inpainting large and irregular hole regions with complex background, which is quite common in consumer electronics. Recent approaches [4], [10] with large or irregular holes has been greatly boosted by advanced deep learning techniques, and a lot of improvements have been proposed. For example, two-stage architectures incorporate additional assistance to handle the image inpainting issue with large hole regions. However, the artifacts produced by the first network would be propagated to the second one. Besides, the dilation rates are uniformly set to a fixed number or even numbers with common divisors, and may lead to gridding artifacts [11].

Recent methods decode low-resolution feature maps using transposed convolution or interpolation, tending to generate undesirable checkerboard artifacts or blurred content lacking high-frequency information. Besides, the dilation operation for extracting features with larger receptive field may lead to

This work was supported by the National Key R&D Program of China under Grant No. 2022ZD0116700, the Science and Technology Planning Project of Zhejiang Province under Grant No. 2022C01090, the Natural Science Foundation of Zhejiang Province under Grant No. LQ21F020015, the Open Research Project Fund of Key Laboratory of Marine Ecosystem Dynamics, Ministry of Natural Resources under Grants No. MED202202, and the National Natural Science Foundation of China under Grant No. 62202131. (*Corresponding author: Yuyu Yin and Honghao Gao.*)

X. Zhang is with School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China; Key Laboratory of Marine Ecosystem Dynamics, Second Institute of Oceanography, Ministry of Natural Resources, China; Hangzhou Dianzi University Shangyu Institute of Science and Engineering, China; School of Computer Science and Engineering, Nanjing University of Science and Technology, China. e-mail: (zhangxin@hdu.edu.cn).

H. Bernd is with Department of Computer Science, University of California, Davis, CA 95616, U.S.A, e-mail: (hamann@cs.ucdavis.edu).

D. Wang, H. Wang, and Y. Yin are with School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, 310018, China. e-mail: (dongjing.wang@hdu.edu.cn, whongbo@hdu.edu.cn, yinyuyu@hdu.edu.cn).

Y. Wang is with the Key Laboratory of Marine Ecosystem Dynamics, Second Institute of Oceanography, Ministry of Natural Resources, Hangzhou, 310012, China (wangyueyun1988@yeah.net)

H. Gao is with School of Computer Engineering and Science, Shanghai University, China and College of Future Industry, Gachon University, Korea (gaohonghao@shu.edu.cn, honghaogao@gachon.ac.kr)

blurry artifacts and semantic inconsistency.

Therefore, we argue that an effective image inpainting in consumer electronics should 1) capture the contextual semantics for generating reasonable image contents, 2) preserve continuity of global structures, and 3) synthesize fine-detail local textures. To achieve these goals, the inpainting network must capture multi-grained information both spatially and semantically. To this end, we propose a Flexible Multi-Grained Dilation Network (FMGDN) for the stated design goals, achieving promising inpainting results.

The backbone of FMGDN is a U-Net with encoder, middle component, and decoder. Different from existing works [12], [13] using dilation convolution, we design a flexible Multi-Grained Residual (MGR) block composed of parallel branches with different dilation rate patterns to capture multi-grained features. More specifically, we use mutual prime numbers instead of even numbers as dilation rates in one branch, which effectively avoids gridding artifacts [11]. The multi-grained features are adaptively integrated with learned channel-specific weights at the end of MGR block, which has three advantages: First, the multi-grained features from parallel branches with different dilation rate patterns in MGR are adaptively integrated with learned channel-specific weights. Second, it can enlarge the receptive field and avoid the gridding artifacts. Third, the feature map resolution of the MGR block remains unchanged, effectively preserving spatial information.

Concerning the decoder part, we introduce a Channel Adaptive Shuffling (CAS) block to re-organize inner elements among channels. Especially, CAS block shuffles elements among channels with channel-specific weights to decode the high-level feature maps back to the image domain. Experiments on three real-world datasets show that FMGDN can inpaint hole regions at a high level of fidelity, handling global, continuous structures as well as fine-detail textures. Especially, the concise and flexible architecture of FMGDN enables modification or configuration to balance between performance and efficiency in real-world applications, which enhances its adaptability to various kinds of consumer electronics.

In summary, our three main contributions are,

- The proposed FMGDN can effective synthesize visually plausible images via filling hole regions with coherent structures and fine texture details learned by one single generative network without additional information;
- The designed Multi-Grained Residual (MGR) block can extract rich multi-grained features with different dilation rate patterns in parallel and adaptive fusion to form deep image semantics for avoiding gridding artifacts;
- The designed Channel Adaptive Shuffling (CAS) block can re-organize inner elements among channels with channel-specific weights and rebuild the inpainted image from high-level image features in a self-adaptive way.

II. Related Work

A. Deep learning-based image inpainting

One-stage architecture. Pathak et al. [4] proposed an encoder mapping the input image to the latent high-level feature space and a decoder building the extracted features up to the image space via up-convolutional layers. However, the results suffer from blurry artifacts and inconsistent boundaries. Yang et al. [14] used pre-trained network to constrain texture consistency and introduce a multi-scale scheme for high-resolution image inpainting. Two discriminators are trained to provide global and local adversarial losses [13], and it needs Poisson blending to alleviate inconsistent boundaries. In addition, partial convolution [15], gated convolution, validness migratable convolution [2], are studied to distinguish between valid pixels and hole pixels.

Multi-stage architecture. Many approaches adopt multi-net architecture to enlarge the receptive field or obtain the support of additional information. Yu et al. [16] designed coarse-to-fine architecture where the coarse generative network predicts the initial result as the input of the fine generative network. The coherent semantic attention method [17] adds skip connections to this architecture to concatenate encoder features and decoder features. Other works use these networks for generating supportive information, such as segmentation prediction [18], edge connection and structure generation [19]. Li et al. [18] combined face parsing network to enable the generative network for consistent contents. Edge connect method generates the completed edge map by one network, and then fills the image holes by another network with the edge map and incomplete image as input. Structure flow [19] divides inpainting task into structure generation and texture synthesis. The two networks in methods [2], [20], one for generating natural images and one for filling the mask images, collaborate for adaptive image inpainting. Multi-net approaches can generate visually plausible images, but need more space and time for parameters and network training.

Attention mechanism. Contextual attention [16] can explicitly reconstruct the hole regions with global background information based on non-local similarity scores. Moreover, more studies on contextual attention includes obtaining attention score by matching low-level feature patches with high-level ones, integrating attention adaptively from multi-scale patches [21] and placing pixel-wise contextual attention layer at multi-stages [22]. Liu et al. [17] proposed coherent semantic attention with correlation between the hole region and the valid region, and utilizes the relationship within the generated patches, effectively facilitating more pixel continuities. Learnable bidirectional attention [20] maps are constructed forward and reversely to allow the network to focus on rebuilding the hole regions with feature re-normalization and mask-updating.

B. Dilated Convolution

Convolution Neural Networks (CNN) and its vaiants [23], such as dilated convolution, are widely applied in computer vision tasks, includes image semantic segmentation [24], [11], [25], deraining [26] and inpainting. The kernel parameters of dilated convolution are distributed with intervals, controlled by the dilation rate. Higher dilation rate indicates larger receptive field. Some inpainting networks [4], [20] use vanilla convolutions with down-sampling operations to enlarge the receptive field, while a dilated convolution can directly obtain large receptive field by inserting holes within the kernels.

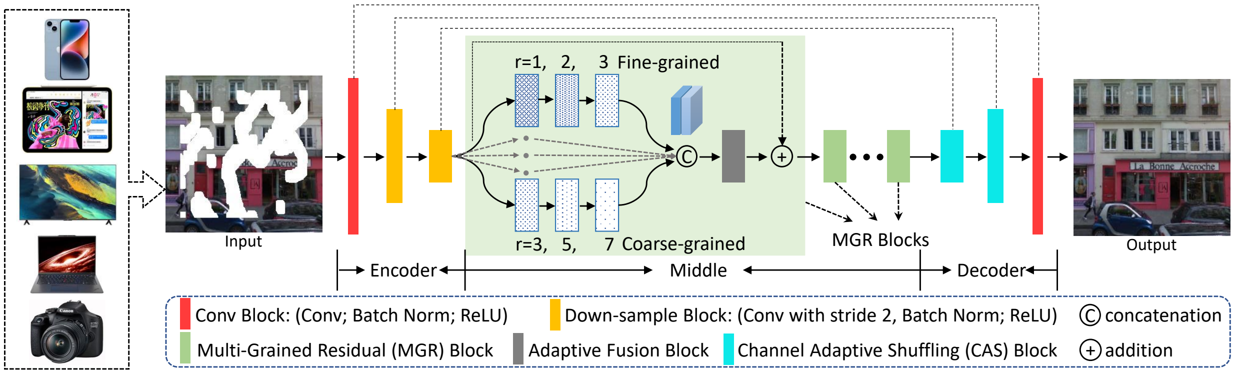

r=1, 2, 3 Fine-grained
r=3, 5, 7 Coarse-grained
Input
Output
MGR Blocks
Encoder
Middle
Decoder
Conv Block: (Conv; Batch Norm; ReLU)
Down-sample Block: (Conv with stride 2, Batch Norm; ReLU)
© concatenation
Multi-Grained Residual (MGR) Block
Adaptive Fusion Block
Channel Adaptive Shuffling (CAS) Block
⊕ addition

Fig. 1. Architecture of FMGDN for image inpainting

Furthermore, some approaches [13], [16] deploy four dilated convolution layers in middle of the encoder-decoder with dilation rates 2, 4, 8, 16, respectively. Meanwhile, other approaches [21], [27] use residual blocks with two dilated convolution layers with dilation rates respectively set as 2 and 1. Hui et al. [12] proposed multi-scale fusion with four parallel dilated convolution layers. Wang et al. [11] proposed hybrid dilated convolution for semantic segmentation where the dilation rates inside one residual block follow a sawtooth wave-like pattern, effectively avoiding gridding artifacts caused by fixed dilation rates.

III. Approach

As shown in Fig. 1, the proposed Flexible Multi-Grained Dilation Network (FMGDN) adopts a one-stage U-Net [28] architecture consisting of the encoder, the middle part and the decoder. The shallower layers (encoder) of U-Net extract low-level features, while the deeper layers focus on capturing high-level semantic information. In particular, long-skip connection between encoder and decoder layers forms typical U-Net architecture to ensure multi-level information well kept in generated images. Specifically, the encoder preliminarily maps the input image to a high-dimensional latent feature space by three convolutional blocks. The middle part further extracts more high-level and hierarchical features by Multi-Grained Residual blocks. The decoder leverages designed Channel Adaptive Shuffling blocks to rebuild images from high-level image feature maps. In particular, FMGDN benefits from its concise and flexible architecture and can be easily configured or simplified for computational power-constrained consumer electronics, such as mobile phones and tablets.

A. Multi-Grained Residual Block

Natural images have rich, multi-grained features. For example, in terms of spatial features, global structures are coarse-grained features, while local textures are fine-grained features. Therefore, effective extraction of multi-grained features is important for synthesizing visually plausible images, and the generated images should maintain reasonable semantics. Therefore, we design Multi-Grained Residual (MGR) block to sufficiently capture multi-grained features and deep semantics. The unit with green background in Fig. 1 shows the pipeline of the MGR block, consisting of 1) the multi-grained feature extraction part and 2) the adaptive fusion part.

The first part of the MGR block use of parallel branches to capture multi-grained features. The number of branches can be flexibly set to adapt to different situations for various kinds of consumer electronics. We used two different branches as an example. The upper branch is fine-grained, and the lower branch is the coarse-grained. Each branch has three dilated convolutional blocks with hybrid dilation rate settings. According to the rules stated in Wang et al. [11], the dilation rates for connected convolution blocks in one group should be mutually prime numbers. We set the three convolutional dilation rates in fine-grained branch to the values 1, 2 and 3, generating a receptive field of 13, to capture fine-scale features. The coarse-grained branch uses dilation rate values 3, 5 and 7, respectively, leading to a larger receptive field of 31 to capture large-scale image patterns. We empirically set dilation rates using the co-prime rule and the principle that branch with the small receptive field captures fine-grained features, and the branch with the large receptive field captures coarse-grained features. Our design of the multi-grained branch unit provides flexibility, as one can easily change the number of branches, number of convolutional layers in each branch, and the dilation rates setting for each branch.

The main differences between MGR block and existing approaches are three-fold. First, we design different parallel branches in MGR block for multi-grained feature extraction. The dilation rates are set under the co-prime principle to avoid artifacts. The branch with lower dilation rates has a smaller receptive field to extract fine-grained and local features. The branch with higher dilation rates has a larger receptive field to extract coarse-grained and global features. Second, the MGR block uses hybrid dilation rates for different branches. The dilation convolution inserts holes into filter kernels, which can be viewed as conducting convolution on different scaled feature maps when different dilation rates are set. Therefore, MGR block extracts multi-scaled features and obtains a larger receptive field. Third, rather than simply adding or concatenating features from different branches, MGR block adaptively fuses multi-grained features with learned weights.

We assume that the input for MGR block is a feature map $I_{in} \in R^{B \times C \times H \times W}$, where B is batch size, C is channel amount, and H and W are spatial height and width. The fine-grained features are obtained by the following equation:

$$I_{fine} = f_{r3}(f_{r2}(f_{r1}(I_{in}))), \tag{1}$$

where f_{r1} is the standard convolutional block, including a convolutional layer with the dilation rate value 1, a batch normalization layer and a rectified linear unit (ReLU) activation layer. The additional layers f_{r2} and f_{r3} have associated dilation rates of 2 and 3, respectively. There is no ReLU layer in f_{r3}. The coarse-grain features are obtained via:

$$I_{coarse} = f_{r7}(f_{r5}(f_{r3}(I_{in}))). \tag{2}$$

The concatenation of fine-grained features I_{fine} and coarse-grained features I_{coarse} in the channel dimension is then provided to the Adaptive Fusion (AF) block. Specfcially, Fig. 2 shows the pipeline of the AF block.

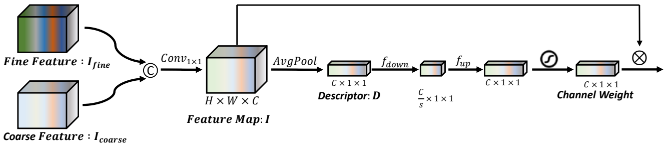


Fig. 2. Adaptive fusion block

First, a 1×1 convolutional operation can assemble multi-grained features to higher-level features and decrease number of channels and parameters for network lightweight, which is crucial for consumer electronics. Feature I is obtained as

$$I = f_{1\times1}(< I_{fine}, I_{coarse} >). \tag{3}$$

Second, the channel-specific features in I contribute differently to the overall goal [29]. We explicitly model the correlation between channels by three steps and re-scale feature I with adaptive channel-specific weights. In step one, the channel-specific descriptor $D \in R^{C\times1\times1}$ is extracted by average pooling operation. The i-th element $d_i \in D$ is obtained by spatially averaging the i-th channel, i.e.,

$$d_i = \frac{1}{H \times W} \sum_{1}^{H} \sum_{1}^{W} I_i. \tag{4}$$

In step two, down-sampling and up-sampling operations can filter out unimportant information, i.e.,

$$D' = f_{up}\left(f_{down}\left(D\right)\right), \tag{5}$$

where re-scaling factor s is 16. In step three, D' is transformed to channel-specific weights via sigmoid function, i.e.,

$$A = f_{sig}(D'). \tag{6}$$

Finally, A is used to re-scale the corresponding input feature map I, and the output of the MGR block is obtained with a residual skip connection [30] as

$$I_{out} = A \otimes I + I_{in}, \tag{7}$$

where $\otimes$ is element-wise multiplication.

B. Channel Adaptive Shuffling Block

In existing decoders: i) transposed convolution in early approaches [4], [14] often cause checkerboard artifacts with uneven overlap [31]; ii) standard convolution with interpolation operations [16] may suppress high-frequency information; interpolation algorithms generally adopt one fixed kernel for the whole image, and cannot adapt to complex conditions; iii) sub-pixel convolution [32] (pixel shuffling), re-organizes feature elements and expands spatial extent by reducing channel dimension; this up-sampling approach shuffles feature elements in a fixed manner.

Then, we designed the Channel Adaptive Shuffling (CAS) block shown in Fig. 3 for self-adaptive feature up-sampling. The low-resolution input feature map for CAS block is I_{low}, representing the concatenation of features from the previous layer and features from the corresponding encoder layer. First, I_{low} is input to convolutional layer to aggregate concatenated features. Second, we perform pixel shuffling for the previous feature map. At this stage, the resolution is scaled up by factor 2 and the channel dimension is reduced by factor 4. To make the shuffled features more adaptive to generate high-fidelity image content, informative channels should be emphasized by being assigned with higher attention scores. Therefore, we adopt the channel re-weight unit introduced in Section III-A. Finally, batch normalization and leaky ReLU is used at the end of CAS block.

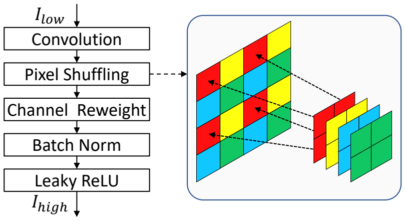


Fig. 3. Pipeline of Channel Adaptive Shuffling block.

C. Loss Function

We combine image-level and feature-level loss functions to optimize the FMGDN. The image-level loss functions include L_1 reconstruction loss for hole regions L_{hole} and valid regions L_{valid}, Multi-scale Structural SIMilarity (MS-SSIM) loss [33] L_{ms} and Total Variation (TV) loss L_{TV}, i.e.,

$$\begin{cases} L_{hole} = \mathrm{E}[||(M - M_{gt}) \otimes (1 - M_m)||_1], \\ L_{valid} = \mathrm{E}[||(M - M_{gt}) \otimes M_m||_1], \\ L_{ms} = \sum_{i=1}^{i=N} W_i(1 - SSIM(P_i^{M_{com}} - P_i^{M_{gt}})), \\ L_{TV} = \mathrm{E}[||M_{com}^{x+1,y} - M_{com}^{x,y}||_1 + ||M_{com}^{x,y+1} - M_{com}^{x,y}||_1], \end{cases} \tag{8}$$

where M, M_{gt}, M_{com} and M_m denote the image generated by FMGDN, the ground truth, the composite image of the generated hole region and original valid region, and the mask image with 1 representing valid region and 0 representing hole region. $\otimes$ is element-wise multiplication. Region-wise L_1 loss aims to preserve pixel-wise fidelity. MS-SSIM loss [34] is based on multi-resolution image pyramid of N levels for better image contrast behavior in high-frequency regions. We adaptively sum the SSIM loss for each pair of image levels, P_i^M and $P_i^{M_{gt}}$, with weight W_i. The level number N and weights for each level W_i adopt previous settings [33]. $\|\|_1$ is L1 Norm (Manhattan norm). TV loss helps to depress gridding and checkerboard artifacts.

The feature-level loss functions include both high-level feature reconstruction loss and the style loss [35], defined as:

$$\left\{\begin{array}{l} L_{feat} = \mathrm{E}[||\phi(M) - \phi(M_{gt})||_1 + ||\phi(M_{com}) - \phi(M_{gt})||_1], \\ L_{style} = \mathrm{E}[||G(\phi(M)) - G(\phi(M_{gt}))||_1 \\ \qquad\qquad +||G(\phi(M_{com})) - G(\phi(M_{gt}))||_1], \end{array}\right. \tag{9}$$

where $\phi(\cdot)$ is feature map from the first three intermediate pooling layers of VGG-16 network [36], pre-trained on ImageNet dataset [37], and $G(\cdot)$ is the Gram matrix of the feature map. Finally, the overall loss function combines the individual losses adaptively as follows:

$$\begin{array}{l} L = \lambda_{hole} L_{hole} + \lambda_{valid} L_{valid} + \lambda_{ms} L_{ms} \\ \quad +\lambda_{TV} L_{TV} + \lambda_{feat} L_{feat} + \lambda_{style} L_{style} \end{array}, \tag{10}$$

where the weights are empirically set as $\lambda_{hole} = 6$, $\lambda_{valid} = 1$, $\lambda_{ms} = 4$, $\lambda_{TV} = 0.1$, $\lambda_{feat} = 0.05$ and $\lambda_{style} = 120$.

Unlike most learning-based inpainting approaches, we do not include adversarial loss that needs to train at least one more discriminative network synchronously with the generative network. Benefitting from specially designed MGR and CAS blocks, and the FMGDN architecture achieves better performance with one generative network, which is important for improving the effectiveness and performance in various kinds of consumer electronics.

IV. Experiments

A. Experimental Design

1) Datasets: We evaluate the proposed FMGDN using three benchmark datasets from real-world. Note that the images in these three different datasets are quite ubiquitous and common in our daily life, so experiments on these dataset may effectively evaluate FMGDN and baselines when applied on consumer electronics, which are ubiquitous and common in our daily life. All images are resized to 256×256.

- Paris StreetView [38] dataset contains 15,000 images collected using vehicles equipped with cameras and sensors.
- CelebA-HQ [39] dataset contains 30,000 high-resolution face images with size of 512×512 from various sources.
- Places2 [40] is a large-scale challenging dataset with 400 scene categories from various online platforms, including photo-sharing websites, social media, and so on.
- Mask dataset [15] consists of 1,2000 different mask images with irregular holes.

2) Comparison methods: We compare the proposed FMGDN with seven state-of-the-art methods, including: (1) **GLCIC** [13] use local/global discriminators for globally and locally consistent image completion; (2) **CA** [16] method, i.e., generative image inpainting with contextual attention and a coarse-to-fine architecture; (3) **Pconv** [15] method, i.e., partial convolution, which conditions the filtering operation on the valid pixels according to the updated mask images; (4) **LBAM** [20] method, i.e., learning bidirectional attention maps for feature re-normalization and mask updating; (5) **RN** [27] use region-wise normalization to separately calculate the statistics in valid region and hole region. (6) **DMFN** [12] uses four-way dilated convolutions as the basic generative block with self-guided regression loss and geometrical alignment constraint. (7) **MADF** [41] use mask-aware dynamic filtering module to learn multi-scale features with an end-to-end cascaded refinement architecture.

3) Settings: Experiments are conducted with PyTorch running on GeForce RTX 3090 GPU. Note that Nvidia GeForce RTX GPUs are popular consumer-grade graphics cards widely used in various kinds of consumer electronics. The training batch size is 16, which can be modified easily to adapt to equiptments with different computing power. We use the Adam algorithm [42] to optimize the FMGDN with parameter values $\beta_1 = 0.5$ and $\beta_2 = 0.9$; the learning rate is 0.0001.

B. Quantitative Comparison

We conduct quantitative evaluation experiments with the proposed FMGDN and the comparison methods in two aspects, including irregular masks and fixed rectangle mask.

Irregular masks. Images with irregular holes are one of the most common cases in photo editing applications of various consumer electronics. Therefore, we quantitatively evaluate FMGDN and the comparison methods on images with different kinds of irregular holes using the metrics PSNR, SSIM [43], L_1, FID [44] and LPIPS [45]. The first three metrics evaluate the differences between completed images and ground truth in pixel domain. The last two metrics are perceptual metrics that measure differences in the deep feature space. FID and LPIPS metrics are considered closer to human visual perception. Statistics obtained on Paris StreetView, CelebA-HQ and Places2 datasets are presented respectively in Table I, Table II and Table III, including the performance data for each mask ratio group and average values for the six mask groups.

From Table I, we observe that the proposed approach FMGDN performs best in all five metrics on Paris StreetView dataset. For metric PSNR, FMGDN gets 29.15 on average, while the second-highest method is MADF with 28.95. For metric SSIM, FMGDN achieves highest value 0.882, while the second highest methods are RN and MADF with 0.880. For metric L_1, FMGDN gets the best value 1.74%, while the second best method MADF gets 1.78%. As Table II shows, on dataset CelebA-HQ, the proposed approach FMGDN gets best performances in terms of metrics PSNR, L_1 , FID and LPIPS. For metric SSIM, FMGDN gets the second highest value 0.902, while MADF gets the highest 0.906. As presented in Table III, FMGDN gets second or third best values in terms of PSNR (our 26.86 with RN 27.04), SSIM (our 0.860 with RN 0.864) and L_1 (our 2.25% with RN 2.18%). However, we get significant advantages for perceptual metrics FID and LPIPS. For metric FID, FMGDN gets best value 58.71, while the second best value is 63.72 by method DMFN. For metric LPIPS, FMGDN gets the lowest value 0.0993, while the second best method DMFN gets 0.1002. In conclusion, our

TABLE I

QUANTITATIVE RESULTS FOR PARIS STREETVIEW DATASET. "↑" AND "↓" INDICATE "HIGHER/LOWER VALUE IS BETTER". BEST AND SECOND BEST RESULTS ARE SHOWN IN BOLD AND UNDERLINE, SEPARATELY.

Metric	Mask ratio	Pconv	LBAM	RN	DMFN	MADF	MGD
PSNR↑	0-10%	35.57	36.83	37.35	37.29	37.75	**37.89**
	10%-20%	30.23	31.22	32.19	31.54	32.22	**32.38**
	20%-30%	27.29	28.15	29.16	28.34	29.10	**29.31**
	30%-40%	25.23	25.94	26.96	26.11	26.89	**27.14**
	40%-50%	23.56	24.18	25.08	24.31	25.13	**25.34**
	50%-60%	21.40	21.86	22.18	21.96	22.60	**22.89**
	Average	27.21	28.02	28.82	28.26	28.95	**29.15**
SSIM↑	0-10%	0.972	0.978	**0.981**	**0.981**	**0.981**	**0.981**
	10%-20%	0.929	0.942	**0.951**	0.949	0.949	**0.951**
	20%-30%	0.898	0.897	**0.913**	0.908	0.910	**0.913**
	30%-40%	0.824	0.848	**0.871**	0.863	0.868	**0.871**
	40%-50%	0.767	0.794	0.822	0.811	0.821	**0.824**
	50%-60%	0.692	0.716	0.745	0.732	0.752	**0.753**
	Average	0.843	0.862	0.880	0.874	0.880	**0.882**
$L_1(\%)$↓	0-10%	0.36	0.30	0.28	0.28	**0.27**	**0.27**
	10%-20%	0.92	0.81	0.71	0.76	0.71	**0.70**
	20%-30%	1.62	1.44	1.27	1.38	1.27	**1.25**
	30%-40%	2.38	2.15	1.90	2.08	1.89	**1.85**
	40%-50%	3.24	2.97	2.67	2.90	2.61	**2.56**
	50%-60%	4.65	4.36	4.17	4.31	3.96	**3.83**
	Average	2.20	2.01	1.83	1.95	1.78	**1.74**
FID↓	0-10%	13.92	8.39	12.44	7.44	11.54	**6.91**
	10%-20%	34.83	21.86	28.31	19.30	31.74	**17.67**
	20%-30%	59.68	38.35	48.63	33.70	60.19	**30.51**
	30%-40%	85.16	55.44	69.71	48.62	91.53	**43.75**
	40%-50%	113.75	74.25	93.24	65.05	129.17	**58.13**
	50%-60%	113.75	74.25	93.24	91.81	190.19	**58.13**
	Average	76.58	50.19	64.02	44.32	85.73	**39.72**
LPIPS↓	0-10%	0.0199	0.0142	0.0196	0.0134	0.0174	**0.0123**
	10%-20%	0.0525	0.0389	0.0511	0.0363	0.0487	**0.0338**
	20%-30%	0.0910	0.0707	0.0886	0.0656	0.0904	**0.0606**
	30%-40%	0.1317	0.1060	0.1279	0.0980	0.1373	**0.0908**
	40%-50%	0.1792	0.1474	0.1763	0.1358	0.1910	**0.1267**
	50%-60%	0.2496	0.2151	0.2643	0.2041	0.2853	**0.1893**
	Average	0.1207	0.0987	0.1213	0.0922	0.1284	**0.0856**

TABLE II

QUANTITATIVE RESULTS FOR CELEBA-HQ DATASET. "↑" AND "↓" INDICATE "HIGHER/LOWER VALUE IS BETTER".

Metric	Mask ratio	Pconv	LBAM	RN	DMFN	MADF	MGD
PSNR↑	0-10%	36.79	38.07	37.64	37.80	**39.19**	39.12
	10%-20%	31.17	32.22	32.84	32.01	**33.37**	33.33
	20%-30%	28.13	29.01	29.70	28.79	30.09	**30.12**
	30%-40%	26.01	26.74	27.32	26.43	27.66	**27.80**
	40%-50%	24.28	24.90	25.28	24.57	25.78	**25.90**
	50%-60%	21.90	22.35	21.99	21.91	23.03	**23.15**
	Average	28.05	28.88	29.13	28.59	29.86	**29.90**
SSIM↑	0-10%	0.977	0.982	0.982	0.982	**0.985**	0.984
	10%-20%	0.942	0.952	0.957	0.953	**0.961**	0.959
	20%-30%	0.901	0.915	0.926	0.916	**0.930**	0.927
	30%-40%	0.859	0.876	0.889	0.876	**0.897**	0.893
	40%-50%	0.814	0.833	0.846	0.831	**0.860**	0.854
	50%-60%	0.753	0.770	0.771	0.764	**0.803**	0.795
	Average	0.874	0.888	0.895	0.887	**0.906**	0.902
$L_1(\%)$↓	0-10%	0.28	0.24	0.26	0.25	**0.21**	0.22
	10%-20%	0.74	0.65	0.61	0.66	**0.56**	0.57
	20%-30%	1.31	1.17	1.09	1.20	**1.02**	1.03
	30%-40%	1.94	1.76	1.65	1.83	1.56	**1.55**
	40%-50%	2.66	2.44	2.36	2.55	2.18	**2.16**
	50%-60%	3.93	3.70	3.97	3.94	3.41	**3.32**
	Average	1.81	1.66	1.66	1.74	1.49	**1.48**
FID↓	0-10%	6.00	3.62	5.35	3.89	4.97	**3.10**
	10%-20%	14.84	9.50	12.65	9.91	13.59	**8.10**
	20%-30%	24.97	16.13	22.20	16.77	25.23	**14.20**
	30%-40%	35.74	23.63	32.20	24.85	39.81	**21.03**
	40%-50%	47.17	31.48	42.61	33.73	56.41	**29.13**
	50%-60%	60.96	41.58	58.81	46.64	83.56	**39.87**
	Average	31.62	20.99	28.97	22.63	37.26	**19.24**
LPIPS↓	0-10%	0.0116	0.0077	0.0124	0.0082	0.0095	**0.0065**
	10%-20%	0.0303	0.0207	0.0320	0.0219	0.0272	**0.0178**
	20%-30%	0.0531	0.0378	0.0558	0.0399	0.0511	**0.0328**
	30%-40%	0.0773	0.0571	0.0819	0.0606	0.0791	**0.0497**
	40%-50%	0.1050	0.0791	0.1131	0.0851	0.1113	**0.0700**
	50%-60%	0.1473	0.1168	0.1755	0.1300	0.1681	**0.1073**
	Average	0.0708	0.0532	0.0785	0.0576	0.0744	**0.0474**

FMGDN generally achieves more stable and better quantitative values compared with five state-of-the-art methods especially under perceptual metrics. The results shown that the proposed FMGDN is effective in capturing semantic information and reconstructing reasonable image contents, and can adapt to various kinds of real-world scenarios.

Fixed rectangle mask. GLCIC [13] and CA [16] use a local discriminator which takes local image patches as input, so it's more suitable to train these two methods with regular masks. Therefore, to make a fair comparison, we train our model FMGDN, GLCIC method, and CA method with fixed center rectangle mask. The quantitative evaluation values with metrics PSNR, SSIM [43], L_1, Fid [44] and LPIPS [45] are listed in Table IV. From Table IV, we can see that FMGDN gets the best values in most cases, except for dateset Paris StreetView. Specifically, FMGDN has advantage in terms of perceptual metrics. The statistics of quantitative measurements further prove that our FMGDN is capable of generating images with high fidelity and visually reasonable contents.

Computational complexity. Furthermore, we measure model complexity with three metrics, including parameter size, time of generating one 256×256 image and MACs (Multiply-Accumulate Operations) in Table V. Note that the parameter size in Table V just refers to the generative network. In terms of processing time, the proposed model FMGDN takes 1.5 ms to generate one 256×256. The least time-consuming method is DMFN using 0.54ms for one image. The most time-consuming method is GLCIC, which takes 214.1 ms for one image with time-consuming Poisson blending as post-processing. In terms of model size, FMGDN has 33.1M trainable parameters, while CA method has minimal parameters, 3.6M, and MADF has maximum parameters, 85.14M. For metric MACs, FMGDN gets the highest value. Although MGD-Net has more multiply-accumulate operations, it only needs 1.5ms to generate one 256 $\times$256 image, which is faster than some methods with lower MACs. We argue that FMGDN is mostly composed of convolutional operations, where GPU can optimize to deal with in parallel. In conclusion, FMGDN has relatively low-level model complexity, but with best perceptual metrics when it comes to the visual image qualities. Moreover, the proposed FMGDN can be configured flexibly to adapt to different situations for various kinds of consumer electronics.

C. Qualitative Results

Irregular masks. Inpainting results on different kinds of images by FMGDN and baselines are shown in Fig. 4. We observe that Pconv [15] basically generates semantically reasonable content, but fails to recover complex structures and textures. For example, in the first two rows, building windows

TABLE III
QUANTITATIVE RESULTS FOR PLACES2 DATASET USING THREE COMPARISON METHODS AND FMGDN. "↑" AND "↓" INDICATE "HIGHER/LOWER VALUE IS BETTER".

Metric	Mask ratio	Pconv	LBAM	RN	DMFN	MADF	MGD
PSNR↑	0-10%	33.41	34.73	35.85	35.37	35.71	**35.89**
	10%-20%	27.91	28.99	**30.37**	29.54	30.02	30.13
	20%-30%	24.95	25.84	**27.22**	26.32	26.88	26.95
	30%-40%	22.93	23.67	**24.98**	24.07	24.69	24.73
	40%-50%	21.31	21.93	**23.15**	22.25	22.92	22.93
	50%-60%	19.26	19.69	**20.65**	19.83	20.55	20.54
	Average	24.96	25.81	**27.04**	26.23	26.80	26.86
SSIM↑	0-10%	0.966	0.973	**0.977**	**0.977**	**0.977**	**0.977**
	10%-20%	0.915	0.930	**0.942**	0.939	0.939	0.940
	20%-30%	0.855	0.876	**0.898**	0.891	0.893	0.894
	30%-40%	0.796	0.820	**0.851**	0.840	0.844	0.845
	40%-50%	0.734	0.760	**0.797**	0.782	0.792	0.791
	50%-60%	0.658	0.678	**0.721**	0.696	0.722	0.715
	Average	0.821	0.839	**0.864**	0.854	0.861	0.860
$L_1(\%)$↓	0-10%	0.46	0.39	**0.34**	0.36	0.35	**0.34**
	10%-20%	1.20	1.04	**0.87**	0.96	0.91	0.91
	20%-30%	2.10	1.86	**1.54**	1.72	1.62	1.62
	30%-40%	3.06	2.76	**2.30**	2.58	2.41	2.40
	40%-50%	4.14	3.79	**3.19**	3.59	3.32	3.31
	50%-60%	5.87	5.52	**4.84**	5.38	4.94	4.92
	Average	2.81	2.56	**2.18**	2.43	2.26	2.25
FID↓	0-10%	17.11	10.68	15.73	9.21	13.36	**8.56**
	10%-20%	44.31	28.75	39.34	24.38	37.76	**22.57**
	20%-30%	78.53	52.79	68.97	44.72	72.59	**41.06**
	30%-40%	113.80	79.41	100.40	67.39	112.19	**62.05**
	40%-50%	153.08	110.40	137.46	94.86	156.50	**87.04**
	50%-60%	201.44	154.70	194.62	141.76	215.04	**130.98**
	Average	101.38	72.79	92.75	63.72	101.24	**58.71**
LPIPS↓	0-10%	0.0226	0.0163	0.0249	0.0145	0.0212	**0.0137**
	10%-20%	0.0597	0.0445	0.0637	0.0390	0.0588	**0.0376**
	20%-30%	0.1049	0.0810	0.1093	0.0707	0.1081	**0.0691**
	30%-40%	0.1512	0.1212	0.1549	0.1060	0.1618	**0.1053**
	40%-50%	0.2019	0.1668	0.2057	0.1476	0.2213	**0.1472**
	50%-60%	0.2753	0.2419	0.2974	0.2237	0.3206	**0.2230**
	Average	0.1359	0.1119	0.1426	0.1002	0.1486	**0.0993**

TABLE IV
QUANTITATIVE RESULTS WITH FIXED CENTER MASK. "↑" AND "↓" INDICATE "HIGHER/LOWER VALUE IS BETTER".

Dataset	Paris StreetView			CelebA-HQ		
Method	GLCIC	CA	MGD	GLCIC	CA	FMGDN
PSNR↑	26.65	23.16	**26.89**	26.10	23.64	**27.22**
SSIM↑	**0.887**	0.846	0.885	0.911	0.868	**0.914**
$L_1(\%)$↓	1.79	2.66	**1.76**	1.69	2.35	**1.49**
FID↓	256.0	111.7	**50.0**	101.0	52.4	**20.5**
LPIPS↓	0.1610	0.1401	**0.0937**	0.0715	0.0759	**0.0400**

in green box by Pconv are blurred and twisted. LBAM [20] can inpaint simple structures and natural faces (second and fourth rows), but cannot preserve intricate and continuous structures. The shape of generated lip in the third row and the line in the last row are not clearly kept. RN [27] synthesizes better structures but produces blurred textures, such as the building windows and trees in the first row. DMFN [12] generates clear structures, but the textures of the roof in the fifth row and the floor in the last row are distorted. MADF [41] keeps well structures, but loses fine-detailed textures. By comparison, our FMGDN generates sharp and consistent structures and fine-detail textures. For example, the seventh column in Fig. 4 including building windows, facial parts, and contents in natural images are synthesized with high fidelity. Moreover, FMGDN captures deep semantic information. For example, in the fourth row, the right eye produced by FMGDN is the same color as the visible left eye, which is in line with the goal of preserving facial symmetry. The other baselines generate brown eyes, while the visible eye is blue. The qualitative visual results show that FMGDN can effectively capture high-level semantics and fill in hole regions with coherent structures and well-ordered details, and the inpainted results can satisfy the requirements of consumer electronics users.

TABLE V
COMPLEXITY COMPARISON. THE LOWEST VALUES ARE MARKED IN BOLD.

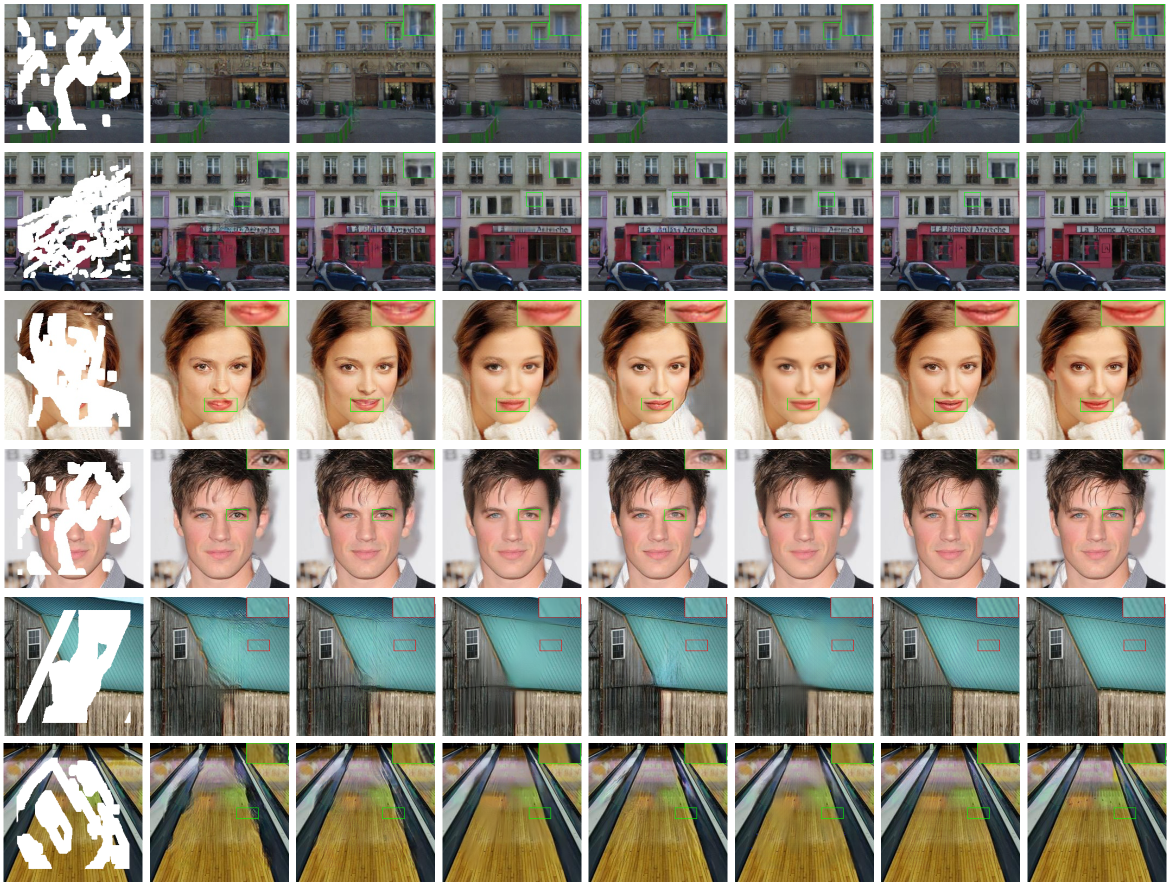
Metric	GLCIC	CA	Pconv	LBAM	RN	DMFN	MADF	FMGDN
Time [ms]	214.1	6.6	0.62	0.76	13.6	**0.54**	5.2	1.5
Parameters [10^6]	6.1	**3.6**	25.78	68.31	11.60	9.04	85.14	33.1
MACs [G]	45.52	22.5	**18.95**	22.11	54.03	58.83	55.51	146.0

Moreover, FMGDN works well for challenging examples with large missing regions. Fig. 5 illustrates inpainting results by FMGDN for images where missing pixels take a large proportion of 40%-60%. As the synthesized images in the third row show, the proposed FMGDN constructs semantically plausible image contents with consistent structures in large missing region situations. The results demonstrate thatFMGDN is effective in capturing global semantics and local detail features. The good inpainting effect is based on our well designed Multi-Grained Residual block and Channel Adaptive Shuffling block, enabling the network to sufficiently capture hierarchical semantic information and effectively decode high-level features to pixel domain self-adaptively, which is important for various real-world applications of consumer electronics.

Fixed rectangle mask. Further experiments are performed with respect to the fixed mask. As shown in the second row of Fig. 6, the centering 128 × 128 pixels (25%) are missed, resulting in relatively few immediate neighboring pixels in center. As shown in the third column of Fig. 6, GLCIC [13] generates blurred image contents and unnatural facial parts, while CA [16] improves the sharpness but the textures are disordered. With well-designed architecture, the proposed FMGDN generates promising results as shown in the fifth column of Fig. 6, with the center region filled with well-ordered textures and sharp structures. For example, in the last row of Fig. 6, FMGDN generates reasonable facial parts and well-structured glasses, instead of missed facial components or only side frame of eyeglasses. The results further show that FMGDN can effectively capture deep semantics and predict globally reasonable pixels with the ability of being aware of contextual information.

D. Ablation Study

MGR block. We train the networks with 2, 4, 6 and 8 MGR blocks using the Paris StreetView dataset. Additionally, we train the architecture, i.e., Branch-1, with just one fine-grain branch. The quantitative performance data is listed in Table VI, which shows that, as the number of MGR blocks increases, the network has better performance. As shown in Fig. 7, the network with more MGR blocks constructs more continuous and clear window structures. Compared with



(a) Input **(b)** Pconv **(c)** LBAM **(d)** RN **(e)** DMFN **(f)** MADF **(g)** FMGDN **(h)** GT

Fig. 4. Qualitative visual comparison. Datasets: rows 1–2 show Paris StreetView images; rows 3–4 show CelebA-HQ images; and rows 5–6 show Places2 images. The columns, from left to right, show: (a) input; (b) Pconv [15] results; (c) LBAM [20] results; (d) RN [27] results; (e) DMFN [12] results; (f) MADF [41] results; (g) our FMGDN results; and (h) ground truth (GT) images. We show zoomed-in patches in the top-right corner via green or red boxes.

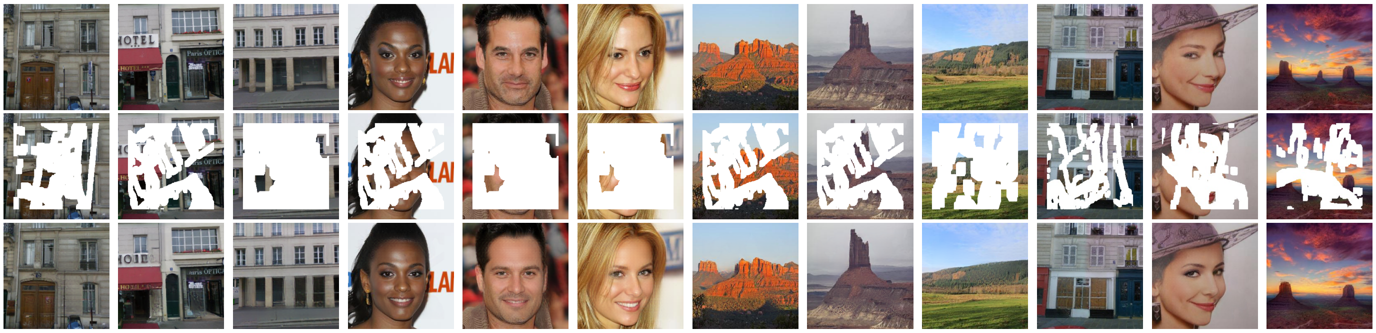


Fig. 5. Image inpainting results for images with large missing regions by the proposed FMGDN. The masks are from the 40%-60% ration section. From top to bottom: ground truth; input images with missing regions marked by white pixels; and inpainting results by FMGDN.

network architecture “Branch-1”, our multi-branch network architecture achieves better performance, both quantitatively and qualitatively. We conclude that the design of multi-grained branches in FMGDN effectively improves image synthesis capability of an image inpainting network.

CAS block. Fig. 8 shows inpainting results with three up-sampling blocks, including transposed convolution, standard convolution with Bilinear interpolation, and our CAS block. We can observe that the transposed convolution intends to produce checkerboard artifacts. Bilinear interpolation produces blurred textures as high-frequency information is lost. CAS block generates images with better visual qualities. This results show that our design of CAS block is effective in generating images with sharp and well-ordered structures.

TABLE VI
Performance of Architectures with Different Numbers of MGR Blocks

MGR	MGR-2	MGR-4	MGR-6	FMGDN	Branch-1
PSNR↑	28.79	28.89	28.97	**29.15**	28.85
SSIM↑	0.878	0.877	0.879	**0.882**	0.877
$L_1(\%)\downarrow$	1.83	1.80	1.78	**1.74**	1.81

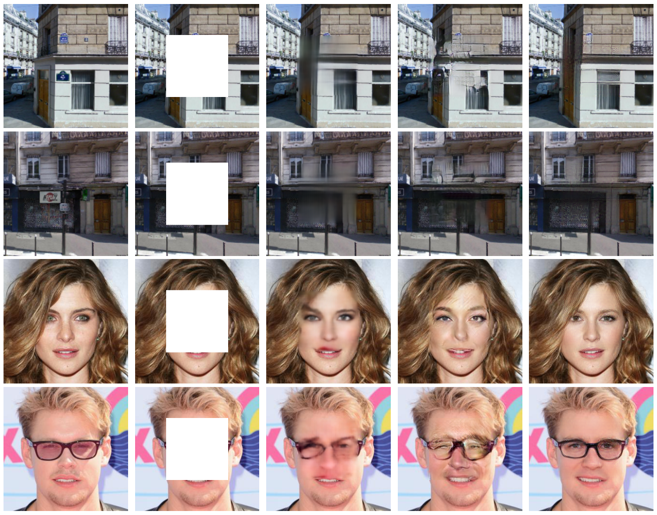


Fig. 6. Image inpainting results with fixed mask. From left to right: ground truth; input images with missing regions marked by white pixels; results by GLCIC [13]; results by CA [16] and results by FMGDN.

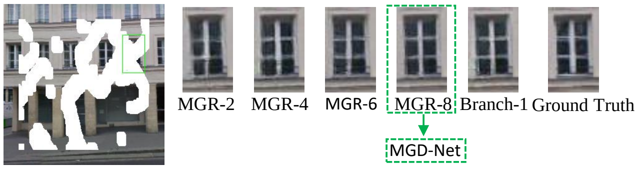


Fig. 7. Results for different MGR block settings.

Branch number. In order to study the effect of different multi-grained branch numbers in the MGR block, we respectively conduct experiments with one branch, two branches, and three branches on Paris StreetView dataset. For the "1-Branch" architecture, the dilation rates are set as 1, 2 and 3, while for the "2-Branch" architecture, the dilation rates are set as 1, 2 and 3 for the fine-grained branch and 3, 5 and 7 for the coarse-grained branch. For the "3-Branch" architecture, we add a coarser-grained branch with dilation rates as 5, 7 and 11 based on the "2-Branches" architecture. As shown in Fig. 9, the "3-Branch" architecture reconstructs the most similar window to the ground truth. The three hybrid branches provide a larger receptive field, which contributes to reconstructing objects with large size. The quantitative evaluation metrics are listed in Table VII, where the architecture with three branches achieves the best performance. Therefore, we conclude that the multi-grained architecture is effective in synthesizing images with multi-scaled textures. The more branches we use, the FMGDN can achieve better results. However, as shown in the last row in Table VII, the architecture with more branches needs more space for parameters. The branch number can be set to adapt to different applications.

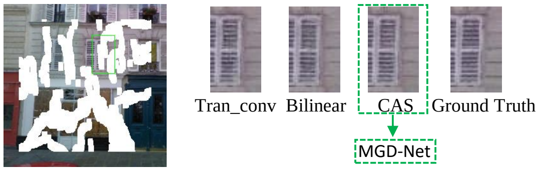


Fig. 8. Results for different up-sampling blocks.

TABLE VII
PERFORMANCE OF ARCHITECTURES WITH DIFFERENT NUMBERS OF MULTI-GRAINED BRANCHES

MGR	1-Branch	2-Branch	3-Branch
PSNR↑	28.85	29.15	**29.23**
SSIM↑	0.877	0.882	**0.884**
L_1(%) ↓	1.81	1.74	**1.72**
Parameters	16.3M	33.1M	47.8M

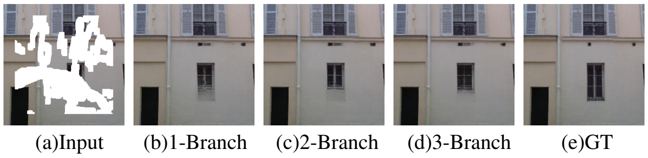

(a)Input (b)1-Branch (c)2-Branch (d)3-Branch (e)GT

Fig. 9. Qualitative visual comparison of FMGDN with different branches.

Dilation setting. For MGR block, we set the dilation rates in different parallel branches employing a co-prime principle to avoid gridding artifacts. For the fine-grained branch, we use relatively small dilation rates for the three sequential convolution layers, such as 1, 2, 3, which works for extracting fine-scaled features. Relatively, the coarse-grained branch should apply larger dilation rates, such as 3, 5, 7. There are various combinations of the dilation rates. To study the effect of different dilation settings, we fix the fine-grained branch with dilation rates 1, 2, and 3. Then, we set a coarse-grained branch with different dilation settings in two independent networks. One is set as 3,5,7, and we name the architecture as "Small". The other is set as 5, 7 and 11, which is named as "Large" architecture. The quantitative evaluations are listed in Table VIII. The "Small" architecture and the "Large" architecture get very close evaluations, where the "Large" architecture obtains better $PSNR$ and $SSIM$ values, while the "Small" architecture gets a better L_1 value. The visual results from these two architectures are shown in Fig. 10, where "Small" architecture reconstructs fine-detailed window textures in the first row marked by green box. For comparison, the "Large" architecture keeps better global structures in the second row. Therefore, we conclude that large dilation rates and small dilation rates have their own advantages and that they can be set flexibly in our FMGDN according to the requirements of real-world applications on consumer electronic devices for balancing bewteen performance and efficiency.

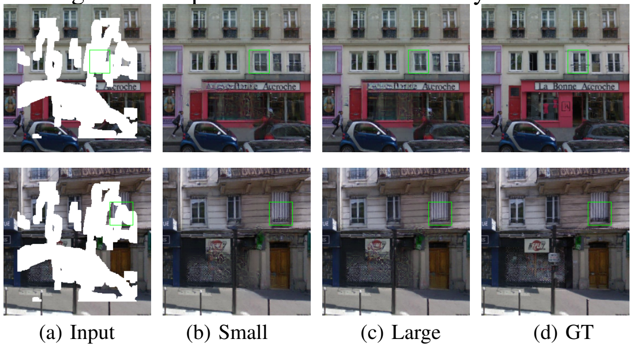

(a) Input (b) Small (c) Large (d) GT

Fig. 10. Qualitative visual comparison of FMGDN with different dilation rate settings.

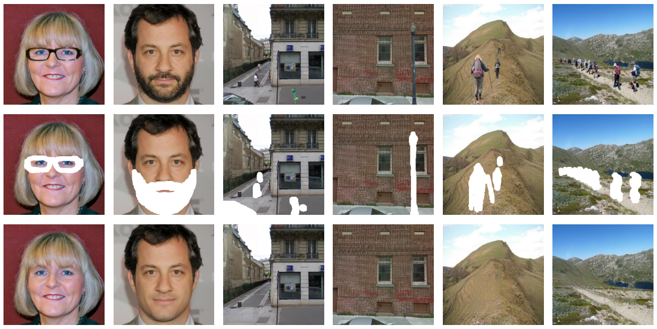


Fig. 11. Results for use cases. Top to bottom: ground truth; input image with removal region specified by user; and inpainting results of FMGDN.

TABLE VIII
PERFORMANCE OF ARCHITECTURES WITH DIFFERENT DILATION RATES

MGR	PSNR↑	SSIM↑	$L_1(\%)\downarrow$
Small	29.15	0.882	**1.74**
Large	**29.18**	**0.883**	1.75

E. Real Use Case

We present six real use cases in Fig. 11, where a user can specify inpainting regions as desired. The first two columns of Fig. 11 show examples of photo editing applications where FMGDN successfully removes the glasses and beard from human faces. In the middle two columns, FMGDN generates clean street view images by replacing specified regions with contextual information. The last two columns show that FMGDN completes the hole regions with plausible natural scenery pixels. The real use cases further demonstrate that FMGDN can be applied to practical cases and synthesize visually plausible content for object removal.

F. User Study

We conduct our user study experiments with the CelebA-HQ dataset and involving 20 volunteers. Each volunteer evaluates 20 random groups of images and votes for the "top two images" exhibiting high visual perceptual qualities, from five images including the ground truth and the results of our MGD-Net and the three comparison methods. The five candidate images are presented in random order. Fig. 12 shows the voting statistics. Except for ground-truth images, MGD-Net obtains the highest number of votes compared with the other three methods. In terms of "top two ranking", the MGD-Net obtains votes comparable to votes for the ground truth, suggesting that MGD-Net generates highly photo-realistic images.

V. CONCLUSION

As for the visual experience enhancement in various kinds of consumer electronics, we have designed a Flexible Multi-Grained Dilation Network (FMGDN) to capture multi-grained information and complete hole regions with semantically and visually plausible contents. Especially, FMGDN benefits from its concise and flexible architecture and can be modified for various kinds of consumer electronics. In future, we will combine our network architecture with advanced attention mechanisms [46] and extend our approach to make it viable for other computer vision tasks. Besides, as for battery-powered consumer electronics, we will try to improve the efficiency of the proposed aproach on consumer electronics based on dynamic resource scheduling method [47], and combine advanced computing techniques [48], [49], including computation offloading and resource allocation techniques [50]–[52], to balance the performance of image inpainting and energy consumption. Besides, we will try to improve the ability of generalization and adaptability for complex application scenarios of image inpainting on consumer electronics based on few-short learning techniques [53].

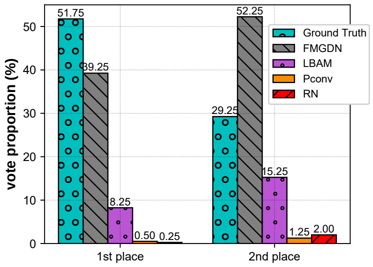


Fig. 12. Voting statistics for top-two images on ground truth and inpainting results from four methods.

REFERENCES

[1] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 9, pp. 10 850–10 869, 2023.

[2] N. Wang, Y. Zhang, and L. Zhang, "Dynamic selection network for image inpainting," *IEEE Transactions on Image Processing*, vol. 30, pp. 1784–1798, 2021.

[3] X. Zhang, B. Hamann, X. Pan, and C. Zhang, "Superpixel-based image inpainting with simple user guidance," in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 3785–3789.

[4] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. Efros, "Context encoders: Feature learning by inpainting," in *Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2536–2544.

[5] C. Yu and L.-Z. Hou, "Realization of a real-time image denoising system for dashboard camera applications," *IEEE Transactions on Consumer Electronics*, vol. 68, no. 2, pp. 181–190, 2022.

[6] M. Kumar and A. K. Bhandari, "Unsupervised enhancement and web tool for perceptually invisible type degraded image," *IEEE Transactions on Consumer Electronics*, vol. 68, no. 4, pp. 401–410, 2022.

[7] Q. Bao, Y. Liu, B. Gang, W. Yang, and Q. Liao, "S^{2}net: Shadow mask-based semantic-aware network for single-image shadow removal," *IEEE Transactions on Consumer Electronics*, vol. 68, no. 3, pp. 209–220, 2022.

[8] T. Yan, H. Li, J. Gao, Z. Wu, and R. W. Lau, "Single image reflection removal from glass surfaces via multi-scale reflection detection," *IEEE Transactions on Consumer Electronics*, vol. 69, no. 4, pp. 1164–1176, 2023.

[9] T. Xu, T.-Z. Huang, L.-J. Deng, X.-L. Zhao, and J.-F. Hu, "Exemplar-based image inpainting using adaptive two-stage structure-tensor based priority function and nonlocal filtering," *Journal of Visual Communication and Image Representation*, vol. 83, p. 103430, 2022.

[10] T. Chen, X. Zhang, B. Hamann, D. Wang, and H. Zhang, "A multi-level feature integration network for image inpainting," *Multimedia Tools and Applications*, vol. 81, no. 27, pp. 38 781–38 802, 2022.

[11] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, "Understanding convolution for semantic segmentation," in *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2018, pp. 1451–1460.

[12] Z. Hui, J. Li, X. Wang, and X. Gao, "Image fine-grained inpainting," *arXiv preprint arXiv:2002.02609*, 2020.

[13] 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.

[14] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, “High-resolution image inpainting using multi-scale neural patch synthesis,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6721–6729.

[15] 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.

[16] 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.

[17] H. Liu, B. Jiang, Y. Xiao, and C. Yang, “Coherent semantic attention for image inpainting,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4170–4179.

[18] 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.

[19] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li, “Structureflow: Image inpainting via structure-aware appearance flow,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 181–190.

[20] C. Xie, S. Liu, C. Li, M.-M. Cheng, W. Zuo, X. Liu, S. Wen, and E. Ding, “Image inpainting with learnable bidirectional attention maps,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8858–8867.

[21] N. Wang, J. Li, L. Zhang, and B. Du, “Musical: Multi-scale image contextual attention learning for inpainting.” in *IJCAI*, 2019, pp. 3748–3754.

[22] N. Wang, S. Ma, J. Li, Y. Zhang, and L. Zhang, “Multistage attention network for image inpainting,” *Pattern Recognition*, vol. 106, p. 107448, 2020.

[23] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian, “A real-time algorithm for signal analysis with the help of the wavelet transform,” in *Wavelets*. Springer, 1990, pp. 286–297.

[24] R. Cong, Y. Zhang, N. Yang, H. Li, X. Zhang, R. Li, Z. Chen, Y. Zhao, and S. Kwong, “Boundary guided semantic learning for real-time covid-19 lung infection segmentation system,” *IEEE Transactions on Consumer Electronics*, vol. 68, no. 4, pp. 376–386, 2022.

[25] C. Schmidt, A. Athar, S. Mahadevan, and B. Leibe, “D2conv3d: Dynamic dilated convolutions for object segmentation in videos,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 1200–1209.

[26] Q. Guo, J. Sun, F. Juefei-Xu, L. Ma, X. Xie, W. Feng, Y. Liu, and J. Zhao, “Efficientderain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, 2021, pp. 1487–1495.

[27] T. Yu, Z. Guo, X. Jin, S. Wu, Z. Chen, W. Li, Z. Zhang, and S. Liu, “Region normalization for image inpainting,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 733–12 740.

[28] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[29] X. Shu, J. Yang, R. Yan, and Y. Song, “Expansion-squeeze-excitation fusion network for elderly activity recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 8, pp. 5281–5292, 2022.

[30] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[31] A. Odena, V. Dumoulin, and C. Olah, “Deconvolution and checkerboard artifacts,” *Distill*, 2016. [Online]. Available: http://distill.pub/2016/deconv-checkerboard

[32] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.

[33] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multiscale structural similarity for image quality assessment,” in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2. Ieee, 2003, pp. 1398–1402.

[34] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, “Loss functions for image restoration with neural networks,” *IEEE Transactions on computational imaging*, vol. 3, no. 1, pp. 47–57, 2016.

[35] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in *European conference on computer vision*. Springer, 2016, pp. 694–711.

[36] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[37] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[38] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. A. Efros, “What makes paris look like paris?” *Communications of the ACM*, vol. 58, no. 12, pp. 103–110, 2015.

[39] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of gans for improved quality, stability, and variation,” in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[40] 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.

[41] M. Zhu, D. He, X. Li, C. Li, F. Li, X. Liu, E. Ding, and Z. Zhang, “Image inpainting by end-to-end cascaded refinement with mask awareness,” *IEEE Transactions on Image Processing*, vol. 30, pp. 4855–4866, 2021.

[42] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[43] 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.

[44] 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,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, vol. 30, 2017, pp. 6626–6637.

[45] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[46] Y. He, X. Jin, Q. Jiang, Z. Cheng, P. Wang, and W. Zhou, “Lkat-gan: A gan for thermal infrared image colorization based on large kernel and attentionunet-transformer,” *IEEE Transactions on Consumer Electronics*, vol. 69, no. 3, pp. 478–489, 2023.

[47] H. Gao, B. Qiu, Y. Wang, S. Yu, Y. Xu, and X. Wang, “Tbdb: Token bucket-based dynamic batching for resource scheduling supporting neural network inference in intelligent consumer electronics,” *IEEE Transactions on Consumer Electronics*, 2023.

[48] V. Hassija, V. Chamola, V. Saxena, V. Chanana, P. Parashari, S. Mumtaz, and M. Guizani, “Present landscape of quantum computing,” *IET Quantum Communication*, vol. 1, no. 2, pp. 42–48, 2020.

[49] M. K. Afzal, Y. B. Zikria, S. Mumtaz, A. Rayes, A. Al-Dulaimi, and M. Guizani, “Unlocking 5g spectrum potential for intelligent iot: Opportunities, challenges, and solutions,” *IEEE Communications Magazine*, vol. 56, no. 10, pp. 92–93, 2018.

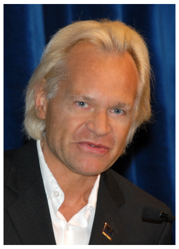
[50] M. Ali, S. Qaisar, M. Naeem, and S. Mumtaz, “Energy efficient resource allocation in d2d-assisted heterogeneous networks with relays,” *IEEE Access*, vol. 4, pp. 4902–4911, 2016.

[51] S. Zhang, H. Gu, K. Chi, L. Huang, K. Yu, and S. Mumtaz, “Drl-based partial offloading for maximizing sum computation rate of wireless powered mobile edge computing network,” *IEEE Transactions on Wireless Communications*, vol. 21, no. 12, pp. 10 934–10 948, 2022.

[52] Z. Zhou, H. Liao, B. Gu, S. Mumtaz, and J. Rodriguez, “Resource sharing and task offloading in iot fog computing: A contract-learning approach,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 3, pp. 227–240, 2019.

[53] J. Xiao, J. Li, and H. Gao, “Fs3dciot: A few-shot incremental learning network for skin disease differential diagnosis in the consumer iot,” *IEEE Transactions on Consumer Electronics*, 2023.

Xin Zhang received the B.S. and Ph.D. degrees in Computer Science and Technology from Shandong University, Jinan, China, in 2012 and 2018, respectively. She was a visitor with the Department of Computer Science, University of California, Davis from September 2016 to August 2017. She is currently a lecturer with Hangzhou Dianzi University, China. Her research interests include deep learning, image processing and computer vision.



Bernd Hamann Bernd Hamann received B.S. degrees in Computer Science and Mathematics from the Technical University of Braunschweig, Germany, in 1985 and 1986, respectively. He received an M.S. degree in Computer Science from the Technical University of Braunschweig, Germany in 1988 and a Ph.D. in computer science from Arizona State University, Arizona, in 1991. His main areas of interest and study are geometric design and computing, data analysis and visualization, and image processing.

Dongjing Wang received the B.S. and Ph.D. degrees in Computer Science from Zhejiang University, Hangzhou, China, in 2012 and 2018, respectively. He was a visitor with the AAI Lab, University of Technology Sydney for one year. He is currently a lecturer with Hangzhou Dianzi University, China. His current research interests include recommender systems, deep learning, and image processing.

Hongbo Wang received his Ph. D. degree from Zhejiang University in 2012. He is currently working as an Associate Researcher, a master's supervisor in the School of Computer Science and Technology, Hangzhou Dianzi University. His main research interests include intelligent information processing, intelligent computing, and image processing.

Yueyun Wang received the Ph.D. degree in Marine Biology from University of Chinese Academy of Sciences, China, in 2017. He is currently an associate researcher at Second Institute of Oceanography, MNR, China. His current research interests include deep-sea biodiversity and marine biological image recognition.

Yuyu Yin received the Ph.D. degree in computer science from Zhejiang University in 2010. He is currently a Professor with the College of Computer, Hangzhou Dianzi University, Hangzhou, China. He is also a Supervisor of master's students with the School of Computer Engineering and Science, Shanghai University, Shanghai, China. He has authored or coauthored more than 40 articles in journals and refereed conferences, such as Sensors, Entropy, IJSEKE, Mobile Information Systems, ICWS, and SEKE. His research interests include service computing, cloud computing, and business process management. Dr. Yin is also a member of the China Computer Federation (CCF) and the CCF Service Computing Technical Committee. He has organised more than ten international conferences and workshops, such as FMSC 2011–2017 and DISA 2012 and 2017–2018. He has served as a Guest Editor for the Journal of Information Science and Engineering and International Journal of Software Engineering and Knowledge Engineering and a Reviewer for the IEEE Transaction on Industry Informatics, Journal of Database Management, and Future Generation Computer Systems.

Honghao Gao is currently with the School of Computer Engineering and Science, Shanghai University, China. He is also a Professor at the College of Future Industry, Gachon University, Korea. Prior to that, he was a Research Fellow with the Software Engineering Information Technology Institute at Central Michigan University, USA, and was an Adjunct Professor at Hangzhou Dianzi University, China. His research interests include Software Intelligence, Cloud/Edge Computing, and Intelligent Data Processing. He has publications in IEEE TII, IEEE T-ITS, IEEE TNNLS, IEEE TSC, IEEE TFS, IEEE TNSE, IEEE TNSM, IEEE TCCN, IEEE TGCN, IEEE TCSS, IEEE TETCI, IEEE/ACM TCBB, ACM TOIT, ACM TOMM, ACM TOSN, ACM TMIS, etc. He is the 2022-2023 recipient of Highly Cited Chinese Researchers by Elsevier, the 2023 recipient of Highly Cited Researcher by Clarivate, and is recognized as World's Top 2% Scientists 2021-2023. Prof. Gao is a Fellow of the Institution of Engineering and Technology (IET), a Fellow of the British Computer Society (BCS), and a Member of the European Academy of Sciences and Arts (EASA). He is the Editor-in-Chief for International Journal of Web Information Systems (IJWIS), Editor for Wireless Network (WINE), The Computer Journal (COMPJ), and IET Wireless Sensor Systems (IET WSS), and Associate Editor for IEEE Transactions on Intelligent Transportation Systems (IEEE T-ITS), IET Intelligent Transport Systems (IET ITS), IET Software, International Journal of Communication Systems (IJCS), Journal of Internet Technology (JIT), and Engineering Reports (EngReports). Moreover, he has broad working experience in cooperative industry-university-research. He is a European Union Institutions-appointed external expert for reviewing and monitoring EU Project, is a member of the EPSRC Peer Review Associate College for UK Research and Innovation in the UK, and a founding member of the IEEE Computer Society Smart Manufacturing Standards Committee.