Uncertainty-Aware Semantic Guidance and Estimation for Image Inpainting

Liang Liao, Jing Xiao, *Member, IEEE*, Zheng Wang, *Member, IEEE*, Chia-Wen Lin, *Fellow, IEEE*, and Shin'ichi Satoh, *Member, IEEE*

***Abstract*—Completing a corrupted image by filling in correct structures and reasonable textures for a complex scene remains an elusive challenge. In case that a missing hole involves diverse semantic information, conventional two-stage approaches based on structural information often lead to unreliable structural prediction and ambiguous visual texture generation. To address the problem, we propose a SEmantic GUidance and Estimation Network (SeGuE-Net) that iteratively evaluates the uncertainty of inpainted visual contents based on pixel-wise semantic inference and optimize structural priors and inpainted contents alternatively. Specifically, SeGuE-Net utilizes semantic segmentation maps as guidance in each iteration of image inpainting, under which location-dependent inferences are re-estimated, and, accordingly, poorly-inferred regions are refined in subsequent iterations. Extensive experiments on real-world images demonstrate the superiority of our proposed method over state-of-the-art approaches in terms of clear boundaries and photo-realistic textures.**

***Index Terms*—Image inpainting, semantic guidance, semantic segmentation, uncertainty estimation.**

I. Introduction

IMAGE inpainting refers to the task of filling a missing area with synthetic content derived from some prior knowledge about the scene. This task has been an active topic in the field of image processing for decades [1]–[5], because it has found a broad range of applications such as object removal, restoration of old films and paintings, image editing, and error concealment in video communication. The key to producing high-quality inpainting results lies in both semantically reasonable contexts and visually pleasing textures [6].

Recently, deep convolutional networks have been applied to address image inpainting problems. Most existing learning-based inpainting methods [7]–[9] resort to an encoder-decoder architecture to infer the context of a corrupted image (Fig. 1(a)) and then refine the textural details on the initial inference of a missing region [10]–[13]. This is based on the assumption that the encoded feature of a corrupted image contains sufficient contextual information for reconstructing the missing region of the image, which is adequate for simple corrupted patterns since the manifold of the context of such patterns can be reasonably well characterized by the encoder network. However, when a corrupted region involves multiple semantic regions, modelling the prior distributions of different semantic categories becomes difficult, since the relationships between the missing pixels and its surroundings become complex. In this case, uniformly mapping different semantics onto a single manifold in the context-based methods often leads to blurry boundaries and incorrect semantic content.

An alternative approach is to infer structural information to assist image inpainting, in which the spatial delineations derived from the inferred structures help alleviate the blurry boundary

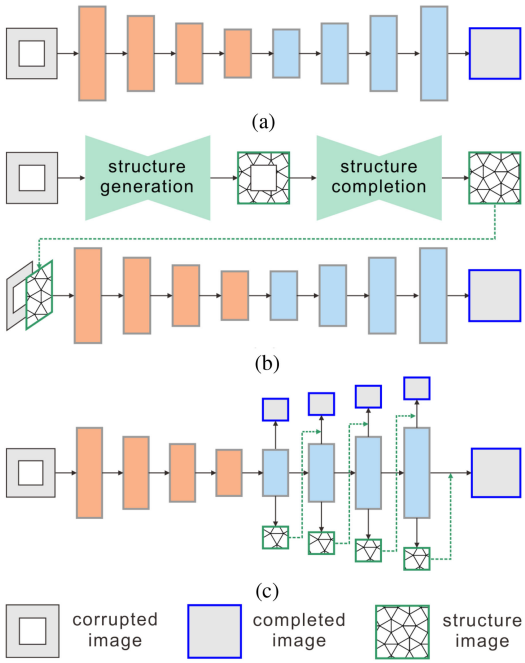


Fig. 1. The architectures of inpainting with structural information. Note that the skip connections are ignored in the sketch. (a) Encoder-decoder without structures; (b) Structures are firstly completed and used as guidance for inpainting mask; (c) Our proposed architecture, in which structure map generation and inpainting tasks are alternatively optimized during decoding.

Manuscript received June 30, 2020; revised October 11, 2020; accepted December 6, 2020. Date of publication December 17, 2020; date of current version February 22, 2021. This work was supported in part by the National Natural Science Foundation of China under Grants 91738302 and 61671336 and in part by the Natural Science Foundation of Jiangsu Province under Grant BK20180234. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Michele Covell. *(Corresponding author: Jing Xiao.)*

Liang Liao and Jing Xiao are with the School of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: liaoliangwhu@whu.edu.cn; jing@whu.edu.cn).

Zheng Wang and Shin'ichi Satoh are with the Digital Content and Media Sciences Research Division, National Institute of Informatics, Tokyo 101-8430, Japan (e-mail: wangz@nii.ac.jp; satoh@nii.ac.jp).

Chia-Wen Lin is with the Department of Electrical Engineering and the Institute of Communications Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan (e-mail: cwlin@ee.nthu.edu.tw).

Digital Object Identifier 10.1109/JSTSP.2020.3045627

problem. This kind of method [14]–[16] is typically based on the two-step architecture illustrated in Fig. 1(b), where structures such as edges or contours are extracted and completed in the first step, followed by completing the corrupted image details guided by the predicted structures in the second step. Nevertheless, these methods ignore the modeling of semantic content, thereby usually resulting in ambiguous textures at the semantic boundaries. Furthermore, the performance of the two-step inpainting process highly relies on the reconstructed structures from the first step, but the uncertainty of the edge or contour connections can largely increase especially when multiple semantic categories appear in the missing region. Additionally, in the second step, simply combining the structural information with the corrupted image as inputs usually cannot provide sufficient guidance for texture generation [17].

To overcome these limitations, we propose to utilize semantic priors [18], [19] in the image inpainting process and develop a new alternative-optimization architecture that progressively updates the semantic guidance and the completed image (Fig. 1(c)) in a coarse-to-fine manner. Specifically, we propose a novel SEmantic GUidance and Estimation Network (SeGuE-Net), in which the segmentation maps are adopted as the semantic guidance. In this way, the object categories, locations, and shapes contained in the segmentation map can provide the semantic boundaries as well as guide the learning of different texture knowledge of diverse semantic categories. Unlike the existing method, namely, SPG-Net [16], using segmentation maps as inpainting guidance in a two-step architecture, our proposed SeGuE-Net makes use of multi-scale guidance on the intermediate predictions at different decoding scales to alternatively optimize the inpainting and segmentation results through their interplay across scales. The encoded image features shared for both tasks can be enforced to contain the semantic and spatial information, which are valuable to learn the prior distributions of different semantic categories.

To further boost the accuracy of the proposed SeGuE-Net on the progressive updating of fill-in pixels, we propose a novel uncertainty estimation mechanism, based on the finding that ambiguous semantic contents usually cannot lead to solid semantic segmentation results. The estimation on semantic uncertainty can effectively locate suspicious uncertain pixels in the previous inpainted regions, so as to adjust the attention for inpainting update in the next iteration. Guided by the estimation result, those wrongly predicted pixels in the previous round of inpainting can potentially be corrected progressively, resulting in inpainting performance improvement.

Our contributions are summarized as follows:

1) We propose an alternative-optimization architecture to exploit how semantic prior can significantly improve the performance of image inpainting for a complex image corruption. The proposed multi-scale interplay between semantic segmentation and image inpainting effectively overcomes several limitations of two-step architectures.
2) We are the first to propose an uncertainty estimation mechanism in image inpainting through semantic segmentation to localize the predicted pixels with uncertain semantic meanings, which enables the inpainting process to correct wrongly predicted contexts and textures progressively.
3) Our model outperforms the state-of-the-art methods in completing multiple missing semantic regions in the sense of generating more realistic semantic contexts and visually pleasing textures.

The remainder of this paper is organized as follows. We introduce related work in Section II. The proposed SeGuE-Net is elaborated in Section III. In section IV, experimental settings and extensive experimental results are presented to demonstrate the superiority of the proposed method and the model analysis is presented in Section V. Finally, conclusions are drawn in Section VI.

II. Related Work

In this section, we briefly review related work in each of the three sub-fields: image inpainting, structural information-guided inpainting, and semantic segmentation.

A. Image Inpainting

Deep learning-based image inpainting approaches [7], [20] are generally based on generative adversarial networks (GANs) to generate the pixels of a missing region. For instance, Pathak *et al.* introduced Context Encoders [7], which was among the first approaches in this kind. The model was trained to predict the context of a missing region, but usually leads to blurry results. Inspired by the concept of Context Encoders, several methods were later proposed to better recover texture details through the use of well-designed loss functions [8], [21], neural patch synthesis [10], residual learning [22], feature patch matching [6], [11], [12], [23], content and style disentanglement [24], and others [9], [25]–[27]. Furthermore, semantic attention was recently proposed in [28] to refine the textures for inpainting. However, most of the above methods were designed for dealing with rectangular holes, but cannot effectively handle large irregular holes. To tackle the problems of inpainting irregular holes, Liu *et al.* [29] proposed a partial convolutional layer, that calculates a new feature map and updates the mask at each layer. Later, Yu *et al.* [13] proposed a gated convolutional layer based on the models in [11] for irregular image inpainting. While these methods perform reasonably well for one category of objects or background, they can easily fail if the missing region contains a mixture of multiple categories of scenes.

B. Structural Information-Guided Inpainting

Recently, structural information has proven to be helpful in assisting image inpainting [30]–[32]. The most outstanding work for bio-signals discrimination is the factorization based method aiming at structural feature extraction of big time series data [27] especially when tackling intensive interferences [26]. These methods are mostly based on two-step networks, where missing structures are reconstructed in the first step and then used to guide the texture generation in the second step. In our previous work [14], edge maps were first introduced as a structural

guide to the inpainting network. This idea was later adopted and improved by Nazeri *et al.* [33] and Li *et al.* [34] in terms of better edge prediction. Similar to edge information, object contours were used by Xiong *et al.* [15] to separately reconstruct the foreground and background areas. Ren *et al.* [35] proposed using smoothed images to carry additional image information other than edges as prior information. Considering semantic information for the modeling of texture distributions, SPG-Net proposed in [16] predicts the semantic segmentation map of a missing region as a structural guidance. The above-mentioned methods show that the structural priors effectively help to improve the quality of the final completed image. However, how to infer correct structures remains challenging, especially when a missing region involves complex semantic structures.

C. Semantic Segmentation

As a widely studied means of inferring semantic contents of an image, Semantic segmentation can predict pixel-level semantic labels. Typical semantic segmentation networks, e.g., FCN [36] and SegNet [37], adopt an encoder to extract semantic features followed by a decoder to upsample the features to predict semantic segmentation labels. In order to improve the segmentation performance, multi-scale features are usually assembled to exploit information from different scales/layers [38], [39]. DeepLab [40] adopts atrous spatial pyramid pooling to encode multi-scale contexts. To iteratively optimize the results, CiSS-Net [41] was proposed to solve the optimization as a markov decision process based on reinforcement learning.

Semantic segmentation has achieved significant progress and can provide semantic priors and scene layouts for image generation [42], making it suitable for deriving semantically structural priors for guiding image inpainting. Nevertheless, the missing part of a corrupted image introduces ambiguity to the segmentation, thereby posing challenges on inferring correct segmentation maps for a missing region, especially when the region involves multiple semantic categories.

III. Semantic Guidance and Estimation Based Inpainting Method

In this section, we describe the proposed image inpainting method based on SeGuE-Net. The proposed method progressively learns the inpainting task and semantic segmentation task in an interleaved manner. Below we first introduce the core idea, and then provide an overview of our network architecture. Next, we dissect individual modules therein and the associated objective function for training SeGuE-Net.

A. Motivation

Our motivation for employing alternative-optimization architecture is mainly based on the assumption that image inpainting and semantic segmentation for a corrupted image can mutually assist each other. Specifically, 1) segmentation maps can provide pixel-wise delineations and semantic labels as guidance to infer missing pixel values, as well as evaluate the already-predicted pixels during the segmentation process; 2) image inpainting can recover the content of missing pixels so as to extract better features for predicting high quality segmentation maps, thereby promoting inpainting quality. Thus our aim is to take the benefit of iterative interplay between the two tasks: recovering the semantic meaning of a corrupted scene and generating realistic textures in the missing area.

As illustrated in Fig. 1(c), the image inpainting and semantic segmentation interplay with each other at each decoding scale. The inpainting process receives the previous coarser-scale segmentation map to conduct feature inference and update the predicted content of missing area, then extract more accurate image features for next finer-scale segmentation. Then the segmentation process takes the new image features to generate a more accurate segmentation probability map, from which it outputs a uncertainty mask as the evaluation result to indicate where should be further refined, together with a segmentation map as the boundary and semantic guidance for the next finer-scale inpainting. Since the segmentation process should be carried out on a complete image, we start with the inpainting block. In this way, the interplay between the two tasks can be progressively modelled into a deep network framework via a coarse-to-fine decoding manner, by which we can make full use of the valuable complementary information between inpainting and segmentation to improve inpainting quality.

B. Overview of Network Architecture

We propose an alternative-optimization architecture to utilize the progressively optimized segmentation maps to improve the inpainting accuracy in a coarse-to-fine manner. The network architecture is illustrated in Fig. 2(a). The encoder (e.g., ResNet) extracts the hierarchical contextual features of the input corrupted image X. Then these features are fed into the decoder to predict the semantic segmentation maps and inpainted images iteratively.

In the decoder, the inpainted contextual features are progressively refined under a multi-scale framework, where the semantic information takes effect in two aspects. First, the confidence scores from the probability maps inferred by the segmentation process is used to identify from the previous inpainting results those suspicious pixels to be further updated in the next inpainting inference. Second, the predicted segmentation maps are involved in the next-scale inference modules to guide the update of the contextual features.

The corrupted image is initially completed in the feature level through a Context Inference Module (CIM) based on the contextual inference method [24]. After that, the decoder gradually maps and refines the inferred contextual features from the coarsest-scale to the finest-scale. Two branches for image inpainting and semantic segmentation are respectively performed based on the contextual features at each scale of the decoder to generate multi-scale completed images $\hat{Y}^L$,..., $\hat{Y}^l$,..., $\hat{Y}^1$ and their corresponding semantic segmentation maps $\hat{S}^L$,..., $\hat{S}^l$,..., $\hat{S}^1$.

$$\begin{cases} \hat{Y}^l = h(\varphi^l) \\ \hat{S}^l = g(\varphi^l), \end{cases} \tag{1}$$

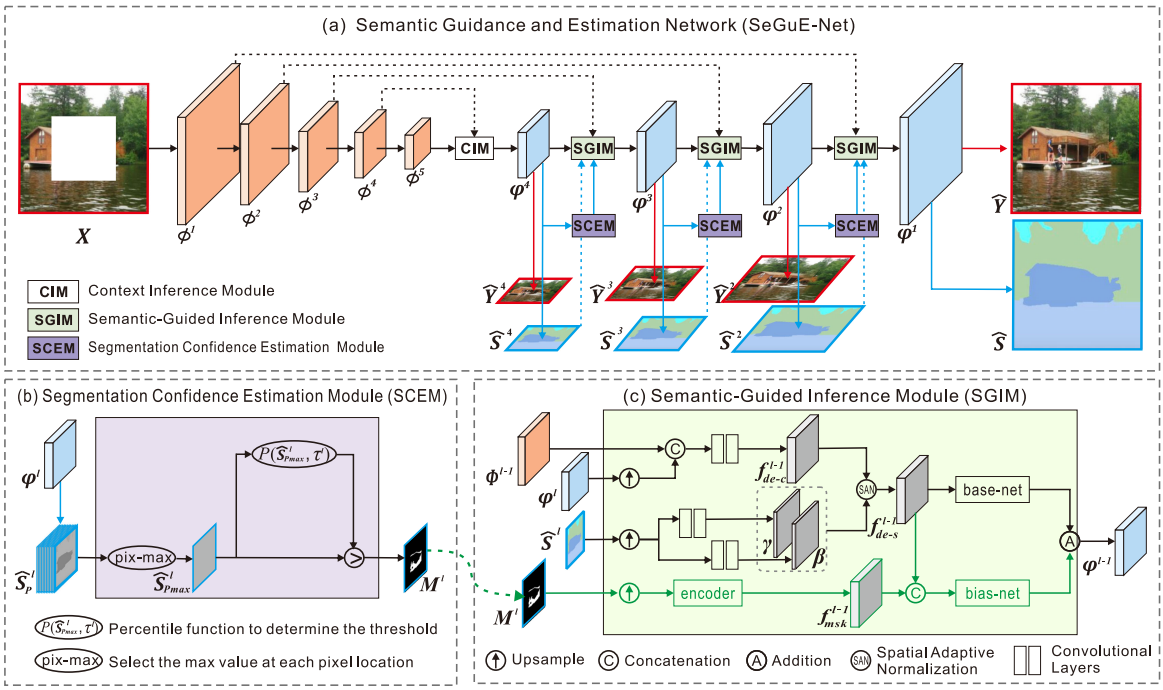


Fig. 2. Overview of the proposed SEmantic GUidance and Estimation Network (SeGuE-Net). It iteratively estimates and updates the contextual features through the SCEM and SGIM modules in a coarse-to-fine manner, where SCEM identifies the pixels where the context needs to be corrected, while SGIM updates the predicted features with the semantic guidance and the uncertainty mask located by SCEM.

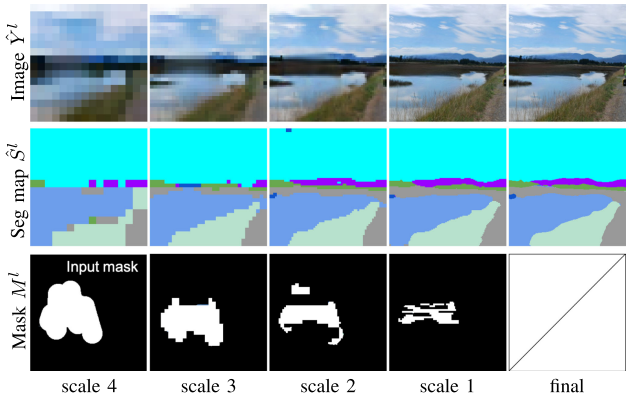


Fig. 3. Visualization of the sample outputs from each scale of SeGuE-Net. From the coarsest scale to the finest scale, the image inpainting and semantic segmentation tasks alternately optimize each other, and the area of uncertainty mask gradually reduces.

where $h(\cdot)$ and $g(\cdot)$ denote the inpainting branch and segmentation branch, respectively. φ^l is the shared contextual feature of the l-th scale in the decoder.

To gradually update the contextual features from an initial corrupted image to its final completed version using semantic information, we propose the semantic uncertainty estimation and semantic-guided inference, that correspond to the Segmentation Confidence Estimation Module (SCEM) and the Semantic-Guided Inference Module (SGIM) in Fig. 2(a), respectively. As illustrated in Fig. 3, from the coarsest scale to the finest scale, the qualities of the inpainted image and the segmentation map are gradually improved, and the area of the uncertainty mask from the SCEM module also reduces as well, owning to the optimized contextual feature.

C. Segmentation Confidence Estimation Module

In traditional structure-completion-first methods, the structures of a corrupted image are completed first, and then the fill-in image contents are directly derived from the completed structures which usually contain incorrectly predicted structures, thereby degrading the inpainting quality. To avoid such flaw, we propose using semantic contexts (i.e., segmentation maps) to alternatively identify those inpainted regions with uncertain contexts that need to be corrected, and then update the semantic contexts, based on an underlying assumption that if meaningless contexts and textures are generated, the segmentation model cannot assign reliable semantic labels to the corresponding pixels.

Specifically, as shown in Fig. 2(b), we deploy a Segmentation Confidence Estimation Module (SCEM) at each decoding scale and introduce a segmentation confidence scoring mechanism to evaluate the inpainted region. We assume that the segmentation branch outputs an intermediate soft prediction $\hat{S}_P^l$ at each pixel as a pixel-wise probability distribution map among the K semantic classes. The class-specific confidence score of a pixel in the map signifies how likely the pixel be attributed to a specific semantic label. The soft prediction is subsequently converted to a max-probability map $\hat{S}_{P\max}^l = \max_{k\in K}\{\hat{S}_P^l\}$ by assigning each pixel with the highest confidence score over the K semantic classes at scale l as the effective confidence score map associated with the prediction.

Based on the confidence score map, an inpainted pixel is considered to have an uncertain semantic label if it has low scores for all semantic classes, resulting in an uncertainty mask. The mask value of a pixel is decided by judging whether the max-confidence score at each pixel location exceeds a threshold,

which is determined by the following percentile function:

$$M^l(p) = \begin{cases} 1, & \hat{S}^l_{P_{\max}} > P(\hat{S}^l_{P_{\max}}, \tau^l) \\ 0, & \text{otherwise} \end{cases}, \tag{2}$$

where $P(\hat{S}_{P_{\max}}, \tau)$ is the percentile function which returns the τth percentile of all pixel values of $\hat{S}_{P_{\max}}$ as the dynamic threshold. τ is a parameter between 0 and 100.

Compared with the soft maps like the segmentation probability map $\hat{S}^l_P$ and confidence score map $\hat{S}_{P_{\max}}$, the binary uncertainty mask $M^l(p)$ defines a clear updating region that indicates those pixels that are likely unreliable. In this way, SCEM enables the model to correct the mistakes in those regions completed at the previous coarser scale. The use of a binary uncertain map simplifies the training of the network compared to the vague clues offered by the soft probability and score maps.

D. Semantic-Guided Inference Module

SGIM is designed to infer and then update the contextual features at the next scale φ^{l-1}. As shown in Fig. 2(c), SGIM takes four types of inputs: two of them are the current contextual features φ^l and the skip features of the next scale ϕ^{l-1} from the encoder. The third is the predicted segmentation map $\hat{S}^l$, which is used to enforce class-specific texture rendering under the assumption that those regions of the same semantic class should have similar textures. The last input is the uncertainty mask M^l. The inference process can be formulated as follows:

$$\varphi^{l-1} = infer(\varphi^l, \phi^{l-1}, \hat{S}^l, M^l). \tag{3}$$

where $infer(\cdot)$ is the process of updating the contextual features in SGIM.

To update the contextual features based on segmentation probability map $\hat{S}^l$, one simple way is to concatenate the segmentation feature and the image feature, but this method has proven to be ineffective in altering the behavior of CNN [17]. In this paper, we follow the image generation approach in [42], which adopts spatial adaptive normalization (SAN) to propagate semantic information to the predicted images for achieving effective semantic guidance. The SAN learns a mapping function $\mathcal{M}$ that outputs a modulation parameter pair (γ, β) based on the semantic prior. The learned parameter pair adaptively influences the inpainting results by applying an affine transformation spatially to each intermediate feature maps at each scale. Specifically, The pair of affine transformation parameters (γ, β) is computed and the contextural features f^{l-1}_{de-s} are updated based on the semantic prior as follows:

$$\begin{cases} \gamma = \mathcal{M}_\gamma(\hat{S}^l) \\ \beta = \mathcal{M}_\beta(\hat{S}^l) \end{cases}, \tag{4}$$

$$f^{l-1}_{de-s} = \gamma \odot \frac{f^{l-1}_{de-c} - \mu}{\sigma} + \beta, \tag{5}$$

where (γ, β) is a pair of affine transformation parameters modeled from segmentation probability map $\hat{S}^l$, μ and σ are the mean and standard deviation of each channel in the concatenated feature vector f^{l-1}_{de-c} generated from ϕ^l and φ^{l-1}. $\odot$ denotes element-wise multiplication. $\mathcal{M}_\gamma$ and $\mathcal{M}_\beta$ are both implemented by two convolutional layers.

Furthermore, in order to correct the pixels classified as 'uncertain' from the SCEM by the uncertainty mask M^l, we propose two sub-networks: base-net F_{ba} and bias-net F_{bi}. The base-net takes f^{l-1}_{de-s} as input and infers a basic context feature, whereas the bias-net is fed with the mask M^l to learn the residuals to rectify the basic context feature. The residuals are computed by concatenating f^{l-1}_{de-s} (what to correct) and the encoded mask feature f^{l-1}_{msk} of M^l (where to correct). We use two convolutional layers to translate the mask into a feature map. The new contextual features at the next scale is formulated as

$$\varphi^{l-1} = F_{ba}(f^{l-1}_{de-s}) + F_{bi}(f^{l-1}_{de-s} \oplus f^{l-1}_{msk}), \tag{6}$$

where $\oplus$ denotes the concatenation operation.

E. Objective Functions

We design appropriate supervised loss terms for learning the inpainting and segmentation tasks at each scale to obtain multi-scale predictions. For image inpainting, we adopt the reconstruction loss to promote the fidelity of a completed image and the adversarial loss to encourage visually realistic fine textures. As for semantic segmentation, we adopt the cross-entropy loss to restrain the distance between the predicted and target class distributions of pixels at each scale. Note that since the segmentation maps do not contain fine textures, the adversarial loss is not required in the segmentation task.

Reconstruction Loss: We use the $\mathcal{L}_1$ loss to encourage per-pixel reconstruction accuracy, and the perceptual loss $\mathcal{L}_p$ to penalizes the discrepancy between the extracted high-level features [21].

$$\mathcal{L}_1(Y, \hat{Y}) = \sum_l \parallel Y - up(\hat{Y}^l) \parallel_1 \tag{7}$$

$$\mathcal{L}_p(Y, \hat{Y}) = \sum_l \sum_{n=1}^{N} \parallel \Psi_n(Y) - \Psi_n(up(\hat{Y}^l)) \parallel_1 \tag{8}$$

$$\mathcal{L}_{re}(Y, \hat{Y}) = \mathcal{L}_l(Y, \hat{Y}) + \lambda_p \mathcal{L}_p(Y, \hat{Y}) \tag{9}$$

where l is the scale, Ψ_n is the activation map of the n-th layer, $up(\cdot)$ is the operation to upsample $\hat{Y}^l$ to the same size as Y, and λ_p is a trade-off coefficient. We use layered features $relu2\_2$, $relu3\_3$, and $relu4\_3$ in VGG-16 [43] pre-trained on ImageNet to calculate the perceptual loss.

Adversarial Loss: As revealed in [9], [44], although the perceptual loss can make the generated image sharper than simply using $\mathcal{L}_1$ loss, high-frequency detailed information is still missing, which can be recovered by adopting the adversarial loss. We use a multi-scale PatchGAN [24] to classify the global and local patches of an image at different scales. The discriminator at each scale is identical and only the input is a differently scaled version of an image. Each discriminator is a fully convolutional PatchGAN which outputs a vector of real/fake predictions, each corresponding to a local image patch. We only use the final completed image $\hat{Y}$ and its ground-truth

image Y to train the discriminator for improving the realisticness of the final completed textures. The adversarial loss is defined as:

$$\mathcal{L}_{ad}(Y,\hat{Y}) = \sum_{t=1,2,3} (E_{p_Y^t \sim Y^t}[\log D(p_Y^t)] + E_{p_{\hat{Y}}^t \sim \hat{Y}^t}[1 - \log D(p_{\hat{Y}}^t)]), \quad (10)$$

where $D(\cdot)$ is the discriminator, p_Y^t and $p_{\hat{Y}}^t$ are the patches in the t-th scaled versions of Y and $\hat{Y}$.

Cross-Entropy Loss: This loss is used to penalize the deviation of $\hat{S}^l$ from the ground-truth labels S at all scales.

$$\mathcal{L}_{se}(S,\hat{S}) = -\sum_{l}\sum_{p\in S} S(p)\log(up(\hat{S}^l)(p)). \quad (11)$$

where p is the pixel index in segmentation map S.

Overall Training Loss: The overall training loss function for our network is defined as the weighted sum of the reconstruction loss, adversarial loss, and cross-entropy loss.

$$\mathcal{L}_{overall} = \mathcal{L}_{re}(Y,\hat{Y}) + \lambda_\alpha \mathcal{L}_{ad}(Y,\hat{Y}) + \lambda_s \mathcal{L}_{se}(S,\hat{S}), \quad (12)$$

where λ_α and λ_s are the weights for the adversarial loss and the multi-scale cross-entropy loss, respectively.

IV. Experimental Comparisons

A. Experimental Settings

1) Datasets: We mainly evaluate the effectiveness of the proposed inpainting method on **Outdoor Scenes** and **Cityscapes** with segmentation annotations. We also extent the test of our trained models on the commonly evaluated datasets for inpainting.

Outdoor Scenes: The dataset [17] consists of 10 200 outdoor scenes image with semantic labels. 9900 of them are used for training and the remaining 300 images for testing. The images are attributed to 8 categories and more than 85% of the dataset are selected from the ADE dataset [45].

Cityscapes: The dataset [46] contains 5000 street view images attributed to 20 categories. In order to enlarge the number of training image of this dataset, we use 2975 images from the training set and 1525 images from the test set for training, and test on the 500 images from the validation set. Since the test set lacks public semantic annotations, we generate them as the ground-truth for training by the state-of-the-art segmentation model Deeplab [47].

We resize each training image to ensure its minimal height/width to be 256 for **Outdoor Scenes** and 512 for **Cityscapes**, and then randomly crop sub-images of size 256×256 as inputs to our model. While training our model, we use common data augmentation strategies, including cropping, scaling, flipping and rotating, to increase the diversity of data. All the inpainting results are directly obtained from our model without any post-processing.

Paris StreetView and Places2: To test the generalization of our trained model on handling images without segmentation annotations, we test the already trained model from **Outdoor Scenes** on **Paris StreetView** and the subset of **Places2** which have similar categories with that of **Outdoor Scenes**. The results can be found in the last part of this section.

2) Baseline Methods: We list all the baselines with their abbreviation and a brief introduction as follows:

GntIpt [11]: Contextual attention for leveraging the surrounding textures and structures, without any auxiliary structural information.

GatedConv [13]: Gated convolution for free-form image inpainting, without any auxiliary structural information.

EdgeConnect [33]: Two-step inpainting with edges as low-level structural information.

SPG-Net [16]: Two-step inpainting with a semantic segmentation map as high-level structural information.

We use GatedConv and EdgeConnect fine-tuned on each dataset and re-implement the model of SPG-Net by ourselves since there is no released model. Since the training of GntIpt assumes availability of the bounding boxes of the holes, which would not make sense for the irregular mask, we directly use their released pre-trained models.

B. Implementation Details

SeGuE-Net is mainly composed of an encoder, a decoder, and CIM between them. The encoder takes 3-channel image and 1-channel mask as input, and gradually down-samples the contextual feature. We build the encoder based on ResNet-50 with five blocks (Conv1, Conv2_x, Conv3_x, Conv4_x, and Conv5_x), which is pre-trained on ImageNet. CIM is used to initially infer, from the contextual features extracted by the encoder, the features for completing an image in the decoder. The decoder gradually updates and refines the inferred contextual features using the SCEM and SGIM from the coarsest scale to the finest scale. To better infer and update the contextual features, we adopt dilated convolution layers to expand the receptive field. At each scale, the inpainting branch consists of two 3×3 convolutional layer for image generation. The segmentation branch generates a K-channel segmentation probability map $\hat{S}_P^l$ after two convolutional layers and a softmax classifier. K is the number of semantic classes.

We implement the SeGuE-Net using the Pytorch toolbox and optimize it and the discriminator using the Adam algorithm with $\beta_1 = 0.5$, $\beta_2 = 0.999$, and a learning rate of 0.0001 following [24]. In all experiments, we use a batch size of 4 and set the training iterations to 500 000. The loss weight λ_p, λ_a and λ_s are set to 1, 0.1, and 5 respectively. The percentile parameters τ^l used to generate the uncertainty mask are set to 75, 50, 25 for scale 2 to scale 4, respectively. Taking the 25 in scale 4, for example, it means the thresholds τ^4 equals to the value, which is greater than 25% of the values in the max-probability map.

C. Performance Comparisons

In this section, we compare our method with the baseline methods from both quantitative and qualitative aspects. We also conduct a user study to assess the quality of the inpainting

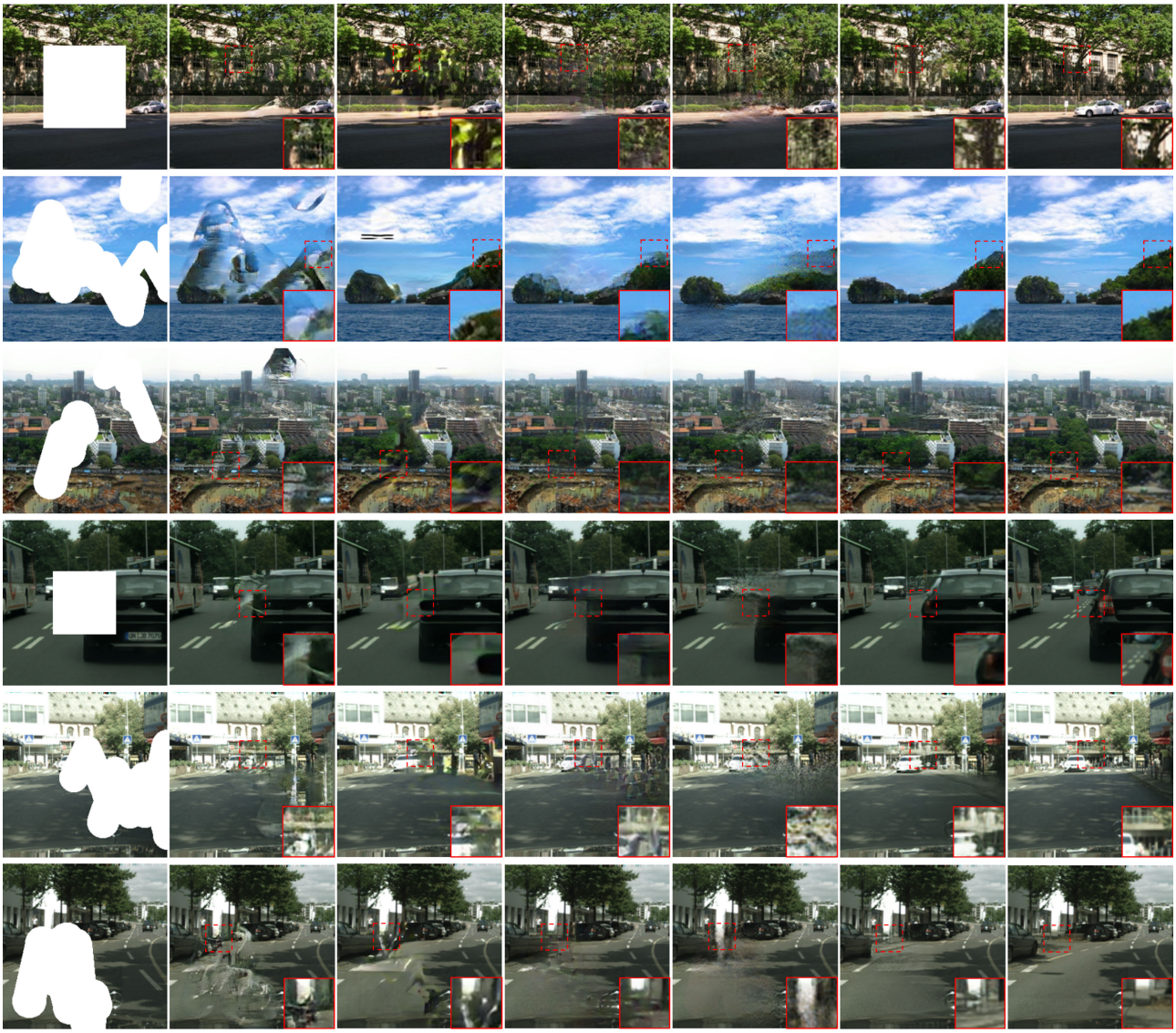


Fig. 4. Qualitative comparisons of inpainting results on image samples from **Outdoor Scenes** (rows 1–3) and **Cityscapes** (rows 4–6).

results, and evaluate the effect of scene complexity on inpainting performance.

1) Qualitative Comparisons: The subjective visual comparisons of the proposed SeGuE-Net with the four baselines (GntIpt, GatedConv, EdgeConnect, SPG-Net) on **Outdoor Scenes** and **Cityscapes** are presented in Fig. 4. The corrupted area is simulated by sampling a central hole (128×128 for **Outdoor Scenes** and 96×96 for **Cityscapes**) or placing masks with random shapes. We use the 12 000 masks from [29] for training and testing. As shown in Fig. 4, the baselines usually generate unrealistic shapes and textures. Obviously GntIpt cannot well handle irregular holes, as evidenced from the many matchless and meaningless textures it produces. GatedConv, EdgeConnect and SPG-Net are effective in handling irregular holes, but unsatisfactory boundaries and over-smooth results in some regions are still often noticeable. In contrast, the proposed method generates more realistic textures and better boundaries between semantic regions than all the baselines, thanks to its semantic guidance and Estimation mechanism.

2) Quantitative Comparisons: We also compare our method quantitatively with the competing methods on the two datasets. Table I shows the numerical results based on three quality metrics: Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) and Fréchet Inception Distance (FID) [48]. FID measures the Wasserstein-2 distance between a ground-truth and its inpainted version using a pre-trained Inception-V3 model [49]: the lower the FID value, the higher the fidelity. Overall, our SeGuE-Net achieves the best objective scores than the baselines, especially in PSNR and SSIM.

3) User Study: In addition to the quantitative and qualitative comparisons, we also conduct a user study based on paired comparisons. we randomly select 100 images from the two datasets (50 from **Outdoor Scenes** and 50 from **Cityscapes**) and invite 30 subjects with image processing expertise to

TABLE I
QUANTITATIVE QUALITY COMPARISON OF FIVE METHODS IN TERMS OF PSNR, SSIM, AND FID ON **OUTDOOR SCENES** AND **CITYSCAPES**
(↑: HIGHER IS BETTER; ↓: LOWER IS BETTER)

	Outdoor Scenes						Cityscapes					
	centering holes			irregular holes			centering holes			irregular holes		
	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓	PSNR↑	SSIM↑	FID↓
GntIpt [11]	18.79	0.73	43.51	17.57	0.72	48.74	20.74	0.73	21.64	16.04	0.62	47.24
GatedConv [13]	19.06	0.73	42.34	19.27	0.81	**40.31**	21.13	0.74	20.03	17.42	0.72	40.57
EdgeConnect [33]	19.32	0.76	41.25	19.63	**0.83**	44.31	21.71	0.76	19.87	17.83	0.73	**38.07**
SPG-Net [16]	18.04	0.70	45.31	17.85	0.74	50.03	20.14	0.71	23.21	16.41	0.67	43.63
SeGuE-Net (ours)	**20.53**	**0.81**	**40.67**	**20.02**	**0.83**	42.47	**23.41**	**0.85**	**18.67**	**18.03**	**0.75**	39.93

TABLE II
QUANTITATIVE PERFORMANCE COMPARISON OF THREE INPAINTING METHODS FOR DIFFERENT SCENE COMPLEXITIES ON 100 IMAGES FROM OUTDOOR SCENES AND CITYSCAPES.

	Low		Moderate		High	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
EdgeConnect [33]	21.42	0.85	20.25	**0.84**	19.43	0.76
SPG-Net [16]	20.82	0.76	20.02	0.72	19.12	0.67
SeGuE-Net (ours)	**21.63**	**0.87**	**20.32**	**0.84**	**19.72**	**0.79**

rank the subjective visual qualities of images completed by five inpainting methods (GntIpt, GatedConv, EdgeConnect, SPG-Net, and our SeGuE-Net). They are not informed of any mask information. For each test image, its five inpainting results are presented in a random order, and each subject is asked to rank the five methods from the best (score: 1) to the worst (score: 5). The result shows that our method receives 54.1% favorite votes (i.e., the top-1 in 1623 out of 3000 comparisons) and average rank of 1.72, largely surpassing 21.4% and 2.42 with EdgeConnect [33], 15.7% and 2.88 with GatedConv [13], 7.3% and 3.64 with SPG-Net [16], and 1.4% and 4.35 with GntIpt [11]. Note, the higher the percentage of favorite votes and the lower the average rank, the better the subjective evaluation. Hence, our method outperforms the other methods.

4) Performance Vs. Scene Complexity: Since our method mainly focuses on completing areas with multiple semantic categories, we also verify its performance on images with different scene complexities. We conduct this analysis by dividing all 100 images used in the user study into three levels of semantic complexities: 1) low-complexity scenes containing 28 images with 1–2 semantic categories; 2) moderate-complexity scenes containing 51 images with 3–4 semantic categories; 3) high-complexity scenes containing 21 images with more than 4 semantic categories. We present the numerical comparisons (see Table II) and visual comparisons (see Fig. 5 and Fig. 6) of SeGuE-Net with two structure-guided baselines (i.e., EdgeConnect and SPG-Net) on the three levels of semantic complexity. The quantitative performance comparison based on PSNR and SSIM in Table II demonstrates that our method outperforms the other methods, especially for high-complexity scenes. Fig. 5 and Fig. 6 also show that the performance gain achieved by SeGuE-Net increases with the scene complexity.

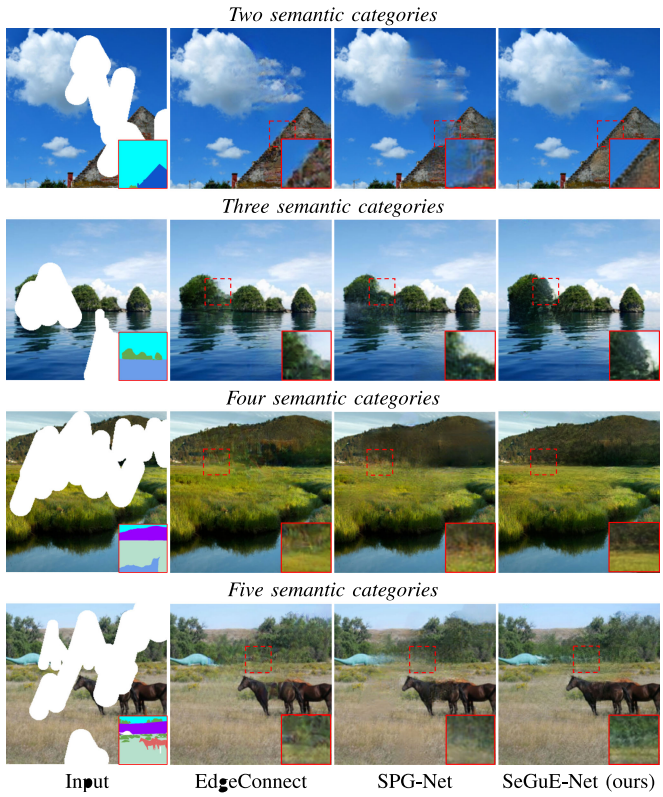


Fig. 5. Qualitative comparisons of test results on image samples with 2 to 5 dominant semantic categories from **Outdoor Scenes**. From left to right: Corrupted image, images completed by EdgeConnect [33], SPG-Net [16] and SeGuE-Net (ours).

5) Results on Paris StreetView and Places2: For a fair comparison with GatedConv and EdgeConnect, we also conduct performance evaluation on **Places2** dataset, which was used for evaluation by both GatedConv and EdgeConnect. Since it contain images with similar semantic scenes to **Outdoor Scenes**, we use our model trained on **Outdoor Scenes** to complete the images with similar scenes in **Places2**. More comparisons with those baselines on **Paris StreetView** are also conducted. The qualitative results in Fig. 7 show that SeGuE-Net is still able to generate proper semantic structures, owing to the introduction of semantic segmentation, which provides better prior knowledge about the scenes.

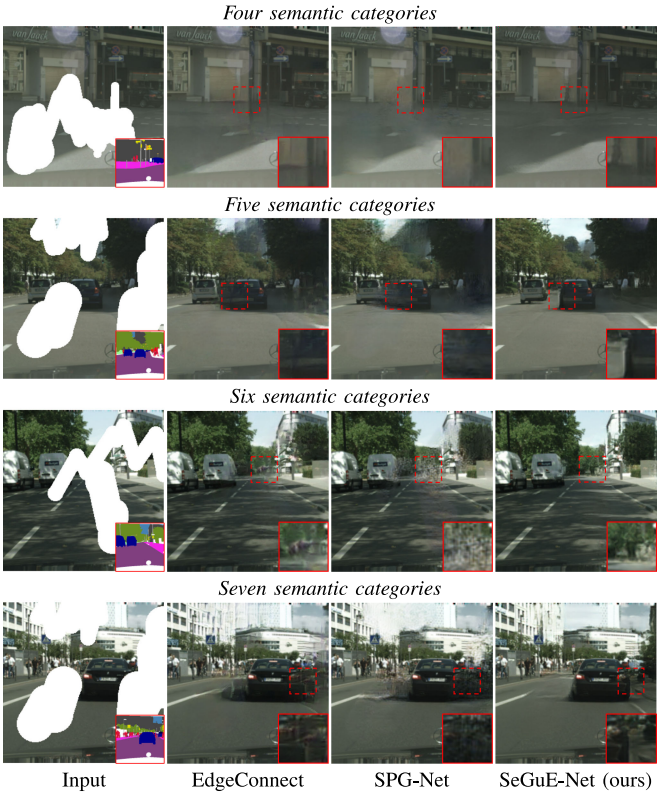


Fig. 6. Qualitative comparisons of test results on image samples of 4 to 7 dominant semantic categories from **Cityscapes**. From left to right: Corrupted image, EdgeConnect [33], SPG-Net [16] and SeGuE-Net (ours).

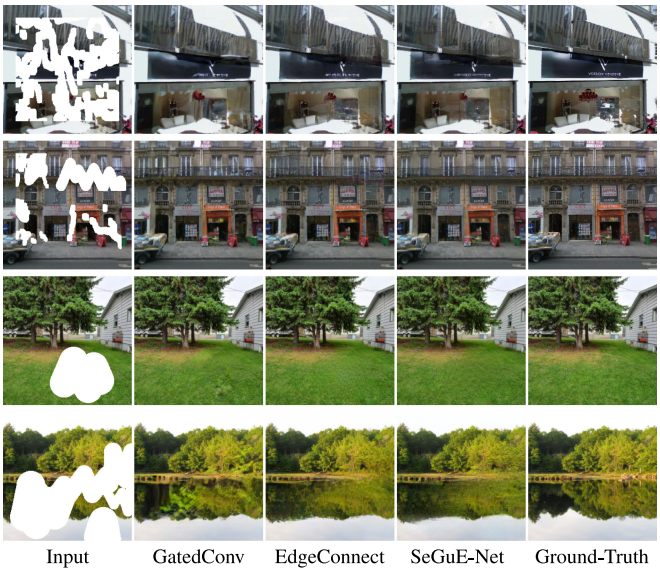


Fig. 7. Qualitative comparisons on image samples from **Paris StreetView** and **Places2**. SeGuE-Net is trained on **Outdoor Scenes** dataset.

V. Model Analysis

A. Ablation Studies

The two core components of the proposed method, uncertainty estimation and semantic-guided inference, are implemented by SCEM and SGIM, respectively. In order to investigate their effectiveness, we conduct an ablation study on three variants: a) Basic-Net (without SCEM and SGIM); b) SG-Net (without SCEM but with the black part of SGIM in Fig. 2); and c) SeGuE-Net (our model with both SCEM and SGIM).

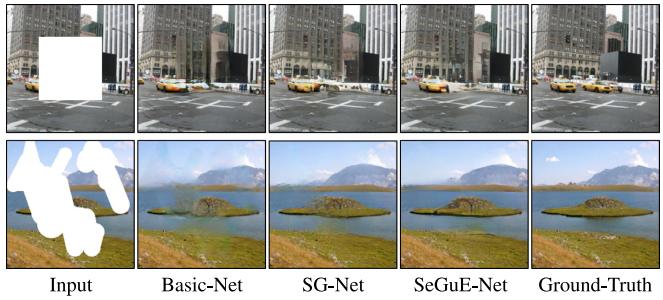


Fig. 8. Qualitative comparisons on three variants to show the effects of SGIM and SCEM.

TABLE III
Quantitative Performance Comparison on the Performances of SGIM and SCEM in Terms of Three Metrics on **Outdoor Scenes** With a Central Hole

	SGIM	SCEM	PSNR↑	SSIM↑	FID↓
Basic-Net	✗	✗	19.14	0.71	43.43
SG-Net	✓	✗	19.58	0.77	41.49
SeGuE-Net	✓	✓	**20.53**	**0.81**	**40.67**

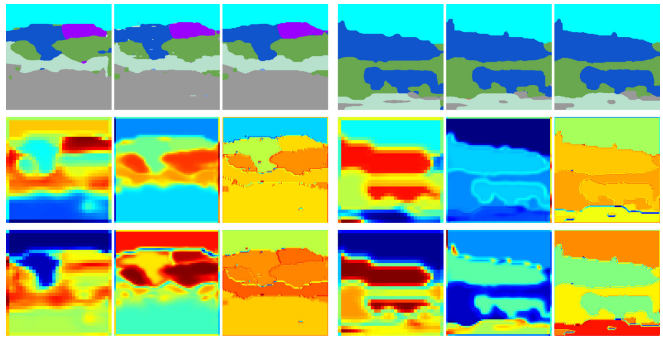


Fig. 9. Visualization of the spatial normalization parameters γ (Row 2) and β (Row 3) based on the segmentation probability (Row 1).

The visual and numerical comparisons on **Outdoor Scenes** dataset are shown in Fig. 8 and Table III. In general, the inpainting performance increases with the added modules. Specifically, the multi-scale semantic-guided interleaved framework does a good job of generating detailed contents, and the semantic segmentation map helps learn a more accurate layout of a scene. With SGIM, the spatial adaptive normalization generates more realistic textures guided by semantic priors. Moreover, SCEM makes further improvements in completing structures and textures (see the fourth column in Fig. 8) by coarse-to-fine optimizing the semantic contents across scales.

Our method modulates the features with semantic prior based on the segmentation map. To explore the correlations between the learned spatial normalization parameters with the segmentation map, we visualize them in Fig. 9. Since the parameters

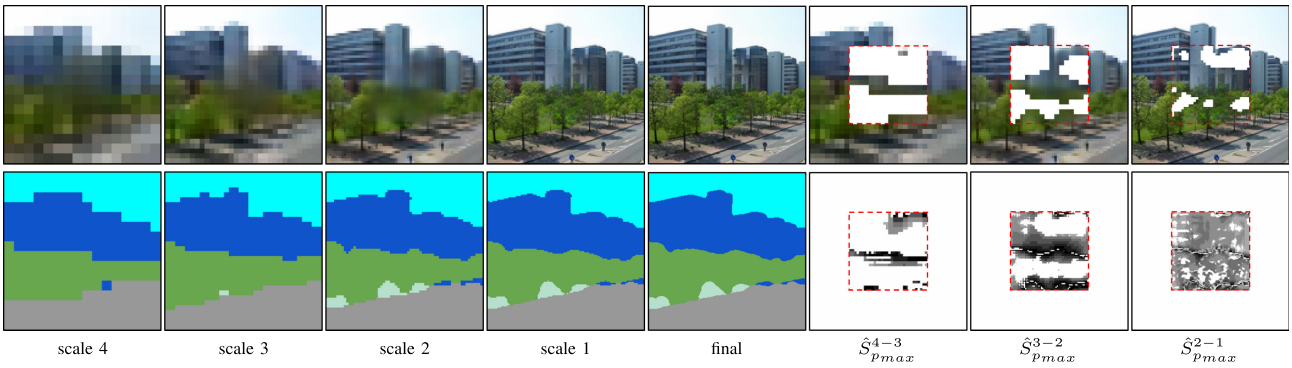


Fig. 10. Illustration of multi-scale progressive refinement with SeGuE-Net. From left to right of the first 5 columns: the inpainted images (top row) and the segmentation maps (bottom row) from scale 4 to scale 1 and the final result. The last 3 columns show the the uncertainty maps (top row) and the confidence score maps (bottom row) of the inpainted region across scales (e.g., $\hat{S}^{4-3}_{P_{\max}}$ shows the confidence score increases from scale 4 to scale 3).

exhibit similar behavior at the same scale, we only show one channel at each scale. As seeing from the heat maps of γ and β pairs, they are closely related to the segmentation map at each scale. It can also be observed that those learned parameters are different in different semantic regions, which can provide semantic priors to the completed regions.

To further verify the effectiveness of SCEM, we visualize a corrupted image and its segmentation maps derived from all decoding scales. As shown in the first five columns of Fig. 10, the multi-scale progressive-updating mechanism gradually refines the detailed textures and the segmentation maps at different scales. The last three columns of the top row show that the region of the uncertainty mask gradually shrinks as well. Correspondingly, the bottom row shows the increase of the confidence scores of segmentation maps from left to right (e.g., $\hat{S}^{4-3}_{P_{\max}}$ showing the increased confidence score from scale 4 to scale 3). The proportion of the white region, which roughly indicates uncertainty labels, also decreases significantly from left to right. The result evidently demonstrates the benefits of SCEM in boosting the semantic correctness of contextual features.

B. *Effectiveness of Uncertainty Estimation*

During the progressive refinement of image inpainting and semantic segmentation, the semantic uncertainty estimation mechanism of SCEM is based on the assumption that the pixel-wise confidence scores from the segmentation probability map can well reflect the correctness of inpainted pixel values. Here we attempt to justify this assumption. Some examples from both datasets are shown in Fig. 11. It can be seen that (except for the confidence scores at the region boundaries):

- The low confidence scores (the white area in row 3) usually appear in the mask area, indicating that the scores reasonably well reflects the uncertainty of inpainted image content;
- the confidence score becomes higher when the scale goes finer, whereas the area of uncertain pixels reduces, meaning that our method can progressively refine the context feature towards correct inpainting.
- the heat maps of updated uncertainty mask features (row 4) show that the mapping features from one channel mask can effectively preserve the distinction between the reliable and uncertain regions. These features function like using the mask as the extra channel with the corrupted image to indicate where should be inpainted.

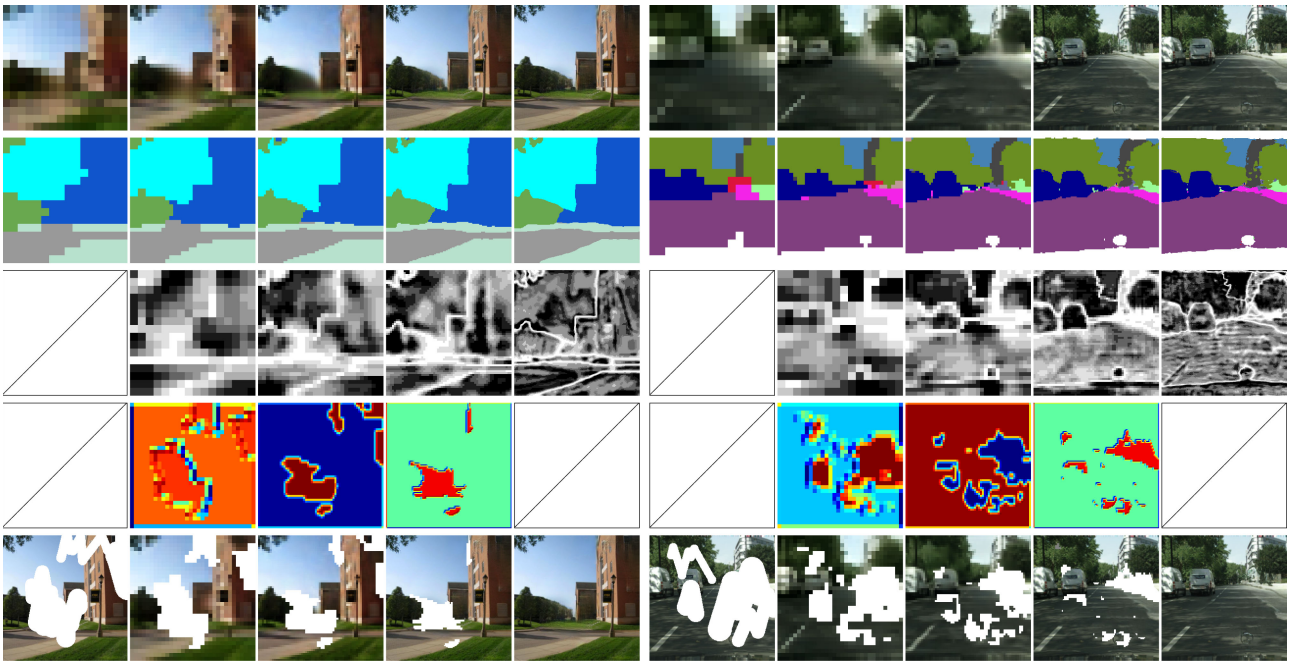
TABLE IV
QUANTITATIVE PERFORMANCE COMPARISON ON MODEL TRAINED BY MACHINE-GENERATED SEGMENTATION (MACH-SEGS) AND HUMAN-LABELED SEMANTICS (LABEL-SEGS) ON **OUTDOOR SCENES**

Outdoor Scenes		Cityscapes	
Methods	PSNR	Methods	PSNR
Mach-segs	20.19	Mach-segs	22.94
Label-segs	**20.53**	Label-segs	**23.41**

We then verify the effectiveness of pixel-wise confidence scores by validating the correlation between the confidence scores of an inpainted image and the $\mathcal{L}_1$ loss with respect to its ground-truth which reflects the fidelity of inpainted pixels. We randomly select 9000 images out of all the training and testing images from the two datasets with center-hole and irregular-hole settings, and calculate the average $\mathcal{L}_1$ loss and the average confidence score of pixels in missing regions. As demonstrated in Fig. 12, the proportion of good-fidelity pixels in each confidence bin generally increases with the segmentation confidence score, implying the confidence score well serves the purpose of evaluating the accuracy of inpainted image.

C. *Impact of Annotations: Human-Labeled Annotations Versus Machine-Generated Segmentation Maps*

For a sanity check, we study the impact of imperfect annotations on the inpainting performance of SeGuE-Net by replacing the human-labeled srmantic annotations for training SeGuE-Net with the maps generated by state-of-the-art segmentation models. We utilize the DPN model [50] pre-trained on [17] and the Deeplab v3+ model [47] as the segmentation tools to generate semantic annotations for **Outdoor Scenes** and **Cityscapes**, respectively. This experiment aims to test the sensitivity of our method to the training samples with imperfect semantic annotations.



(a) **Outdoor Scenes** (b) **Cityscapes**

Fig. 11. Correspondence between the confidence score value and the uncertainty of inpainted image content. (a) **Outdoor Scenes** and (b) **Cityscapes**. Row 1: Inpainted image. Row 2: Predicted segmentation map. Row 3: the confidence score map (darker color means higher confidence score, and vice versa). Row 4: visualization of the encoded uncertainty mask feature in SGIM. Row 5: uncertain pixel map (white pixels indicate uncertain pixels). Since the map at scale 4 is the same as the input mask, we put the input image for better comparison.

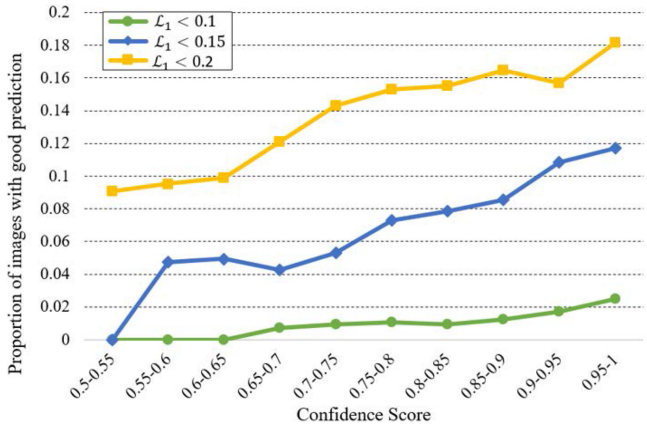


Fig. 12. Correlation between inpainting quality and confidence score.

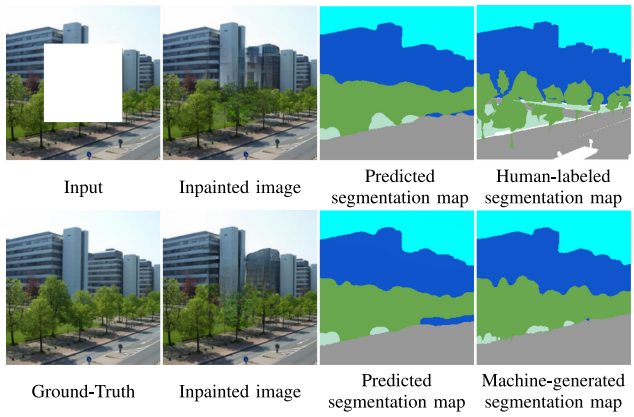


Fig. 13. Impact of training with human-labeled vs. machine-generated segmentation maps on **Outdoor Scenes**. Row 1: using human-labeled segmentation maps; Row 2: using segmentation maps generated by DPN model.

As shown in Table IV, the performance degradation on SeGuE-Net due to imperfect semantic annotations is not significant, meaning that our model can still do a fairly good job even trained on machine-generated semantic annotations. Fig. 13 shows the visual quality comparisons of the inpainting results with SeGuE-Net trained on the human-labeled segmentation maps and on the machine-generated maps. As can be observed, SeGuE-Net trained on imperfect semantic annotations achieves comparable inpainting performance with SeGuE-Net trained on human-labeled semantics. Note that the semantic annotations, either human-labeled or machine-generated, are only used in the training stage of our model. While completing an image, SeGuE-Net itself can automatically generate the inpainted image and segmentation map simultaneously, without the need of semantic annotations.

D. Comparison of Segmentation Accuracy Between SeGuE-Net and Segmentation-After-Inpainting

The success of semantics-guided inpainting largely relies on the quality of inferred semantic labels. SeGuE-Net utilizes multi-scale iterative interleaving of inpainting and semantic segmentation to improve the accuracy of the semantic segmentation

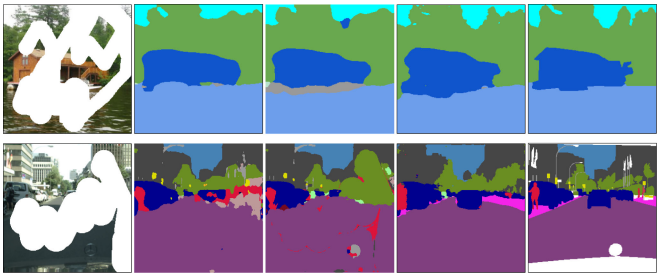


Fig. 14. Visual comparisons on semantic segmentation between SeGuE-Net and the segmentation-after-inpainting solutions.

TABLE V
STATISTICAL COMPARISON ON SEMANTIC SEGMENTATION ACCURACY BETWEEN SEGUE-NET AND THE SEGMENTATION-AFTER-INPAINTING SOLUTIONS ON **OUTDOOR SCENES** AND **CITYSCAPES**

Outdoor Scenes		Cityscapes	
Methods	mIoU%	Methods	mIoU%
EdgeConnect +DPN	0.62	EdgeConnect +Deeplab	0.48
SPG-Net+DPN	0.56	SPG-Net+Deeplab	0.46
SeGuE-Net (ours)	**0.68**	**SeGuE-Net (ours)**	**0.53**

map for a corrupted image. Here we conduct experiments to validate whether the iterative interleaving of inpainting and segmentation outperforms the traditional non-iterative segmentation-after-inpainting strategy in semantic segmentation accuracy. We compare the segmentation maps generated by SeGuE-Net with their counterparts extracted from images completed by a baseline inpainting method.

Fig. 14 and Table V show that SeGuE-Net evidently beats the segmentation-after-inpainting method since SeGuE-Net leads to more accurate semantic assignments and object boundaries, thanks to its multi-scale alternative-optimization of semantics and image contents.

E. Effect of Image Resolution

We also evaluate our model for the inpainting of high resolution (HR) images. The first row of Fig. 15 shows the inpainting performance on images of the original size in **Outdoor Scenes** test set and the second row shows the HR image inpainting of **Cityscapes** with 1024×512. The result shows the effectiveness of our model on HR images.

F. Limitations

Fig. 16 shows some typical failure cases of our model. In general, the most common failure cases are on the non-rigid bodies, e.g., a person or an animal, where the semantic object shapes are hard to learn by the model. For example, in the first row, these models wrongly reconstruct the wall on the lion's head, leading to a poor completion. In the second row, one hand of the second person is missing, but the main part of the body is completed well by our model.

Fig. 15. Results on high resolution images. The image size from the top to bottom: 384×288, 768×512 and 1024×512.

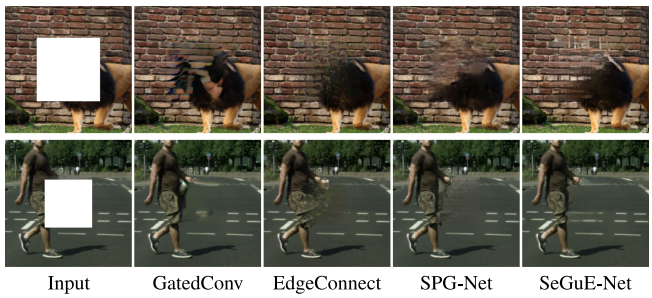


Fig. 16. Examples of failure cases from **Outdoor Scenes** and **CityScapes**.

VI. CONCLUSION

We proposed a novel semantic segmentation guided scheme to complete corrupted images of mixed semantic regions. To address the problem of uncertain semantic segmentation due to missing regions, we have proposed a multi-scale alternative optimization mechanism to conduct interplay between semantic

segmentation and image inpainting. Extensive experimental results demonstrate that the proposed mechanism can effectively refines poorly-inferred pixels through segmentation confidence estimation to generate promising semantic structures and texture details in a coarse-to-fine manner.

REFERENCES

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. Conf. Comput. Graph. Interact. Tech.*, 2000, pp. 417–424.

[2] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[3] C. Guillemot and O. Le Meur, "Image inpainting: Overview and recent advances," *IEEE Signal Process. Mag.*, vol. 31, no. 1, pp. 127–144, Jan. 2014.

[4] Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1153–1165, May 2010.

[5] M. Ghorai, S. Samanta, S. Mandal, and B. Chanda, "Multiple pyramids based image inpainting using local patch statistics and steering kernel feature," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5495–5509, Nov. 2019.

[6] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan, "Shift-net: Image inpainting via deep feature rearrangement," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[7] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2536–2544.

[8] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 107:1–107:14, 2017.

[9] C. Wang, C. Xu, C. Wang, and D. Tao, "Perceptual adversarial networks for image-to-image transformation," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 4066–4079, Aug. 2018.

[10] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, "High-resolution image inpainting using multi-scale neural patch synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4076–4089.

[11] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* , 2018, pp. 5505–5514.

[12] Y. Song *et al.*, "Contextual-based image inpainting: Infer, match, and translate," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–18.

[13] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 4470–4479.

[14] L. Liao, R. Hu, J. Xiao, and Z. Wang, "Edge-aware context encoder for image inpainting," in *Proc. IEEE Conf. Acoust. Speech, Signal Process.*, 2018, pp. 3156–3160.

[15] W. Xiong *et al.*, "Foreground-aware image inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5840–5848.

[16] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, and C.-C. J. Kuo, "Spg-net: Segmentation prediction and guidance network for image inpainting," in *Proc. Brit. Mach. Vis. Conf.*, 2019, p. 97.

[17] X. Wang, K. Yu, C. Dong, and C. Change Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 606–615.

[18] L. Liao, J. Xiao, Z. Wang, C.-W. Lin, and S. Satoh, "Guidance and evaluation: Semantic-aware image inpainting for mixed scenes," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 683–700.

[19] J. Chen, J. Chen, Z. Wang, C. Liang, and C.-W. Lin, "Identity-aware face super-resolution for low-resolution face recognition," *IEEE Signal Process. Lett.*, vol. 27, pp. 645–649, 2020.

[20] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6882–6890.

[21] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 694–711.

[22] U. Demir and G. Unal, "Deep stacked networks with residual polishing for image inpainting," 2017, *arXiv:1801.00289*.

[23] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5961–5970.

[24] J. Xiao, L. Liao, Q. Liu, and R. Hu, "CISI-net: Explicit latent content inference and imitated style rendering for image inpainting," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 354–362.

[25] Q. Wang, H. Fan, L. Zhu, and Y. Tang, "Deeply supervised face completion with multi-context generative adversarial network," *IEEE Signal Process. Lett.*, vol. 26, no. 3, pp. 400–404, Mar. 2019.

[26] D. Chen, Y. Tang, H. Zhang, L. Wang, and X. Li, "Incremental factorization of big time series data with blind factor approximation," *IEEE Trans. Knowl. Data Eng.*, early access, 2019, doi: 10.1109/TKDE.2019.2931687.

[27] D. Chen, Y. Hu, L. Wang, A. Y. Zomaya, and X. Li, "H-parafac: Hierarchical parallel factor analysis of multidimensional big data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 4, pp. 1091–1104, Apr. 2017.

[28] H. Liu, B. Jiang, Y. Xiao, and C. Yang, "Coherent semantic attention for image inpainting," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 4169–4178.

[29] 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 *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 89–105.

[30] J. Sun, L. Yuan, J. Jia, and H.-Y. Shum, "Image completion with structure propagation," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 861–868, 2005.

[31] P. Buyssens, O. Le Meur, M. Daisy, D. Tschumperlé, and O. Lézoray, "Depth-guided disocclusion inpainting of synthesized rgb-d images," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 525–538, Feb. 2017.

[32] J. Liu, S. Yang, Y. Fang, and Z. Guo, "Structure-guided image inpainting using homography transformation," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3252–3265, Dec. 2018.

[33] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, " EdgeConnect: Structure guided image inpainting using edge prediction," in *Proc. Int. Conf. Comput. Vis. Workshop*, 2019, pp. 3265–3274.

[34] J. Li, F. He, L. Zhang, B. Du, and D. Tao, "Progressive reconstruction of visual structure for image inpainting," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 5962–5971.

[35] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li, " StructureFlow: Image inpainting via structure-aware appearance flow," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 181–190.

[36] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[37] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[38] G. Lin, F. Liu, A. Milan, C. Shen, and I. Reid, "Refinenet: Multi-path refinement networks for dense prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1228–1242, May. 2020.

[39] A. Mustafa, H. Kim, and A. Hilton, "Msfd: Multi-scale segmentation-based feature detection for wide-baseline scene reconstruction," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1118–1132, Mar. 2019.

[40] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[41] Y. Zhou, X. Sun, Z.-J. Zha, and W. Zeng, "Context-reinforced semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4046–4055.

[42] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2337–2346.

[43] A. Z. K. Simonyan, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Rep.*, 2015, pp. 1–14.

[44] M. S. M. Sajjadi, B. Schölkopf, and M. Hirsch, "Enhancenet: Single image super-resolution through automated texture synthesis," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 4501–4510.

[45] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20 k dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 633–641.

[46] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.

[47] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.

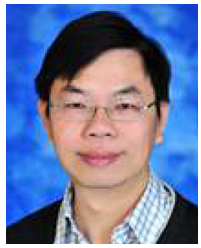
[48] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, " GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *Proc. Adv. Neural Inf. Process.*, 2017, pp. 6626–6637.

[49] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.
[50] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, "Deep learning markov random field for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1814–1828, Aug. 2018.

Liang Liao received the B.S. degree from the International School of Software, Wuhan University, Wuhan, China, in 2013, and the Ph.D. degree from the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan, China, in 2019. He is currently a Researcher with the Digital Content and Media Sciences Research Division, National Institute of Informatics, Tokyo, Japan. His research interests include image processing and transmission.

Jing Xiao (Member, IEEE) received the B.S. and M.S. degrees from Wuhan University in 2006 and 2008, respectively, and the Ph.D. degree from the Institute of Geo-Information Science and Earth Observation, Twente University, The Netherlands, in 2013. She is currently an Associate Professor with the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University. She is also a Project Researcher with the National Informatics Institute, Japan. Her research interests include image/video processing and compression and analysis.

Zheng Wang (Member, IEEE) received the B.S. and M.S. degrees from Wuhan University, Wuhan, China, in 2006 and 2008, respectively, and the Ph.D. degree from the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan, China, in 2017. He is currently a JSPS Fellowship Researcher with Shin'ichi Satoh's Lab, National Institute of Informatics, Japan. His research interests focuses on person re-identification and instance search. He received the Best Paper Award at the 15th Pacific-Rim Conference on Multimedia (PCM 2014) and the 2017 ACM Wuhan Doctoral Dissertation Award.



Chia-Wen Lin (Fellow, IEEE) received the Ph.D. degree in electrical engineering from National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2000.

He is currently Professor with the Department of Electrical Engineering and the Institute of Communications Engineering, NTHU. He is also Deputy Director of the AI Research Center of NTHU. He was with the Department of Computer Science and Information Engineering, National Chung Cheng University, Taiwan, during 2000–2000. Prior to joining academia, he worked for the Information and Communications Research Laboratories, Industrial Technology Research Institute, Hsinchu, Taiwan, during 1992–1992. His research interests include image and video processing, computer vision, and video networking.

Dr. Lin was a Distinguished Lecturer of IEEE Circuits and Systems Society from 2018 to 2019, a Steering Committee member of IEEE TRANSACTIONS ON MULTIMEDIA from 2014 to 2015, and the Chair of the Multimedia Systems and Applications Technical Committee of the IEEE Circuits and Systems Society from 2013 to 2015. His articles received the Best Paper Award of IEEE VCIP 2015, Top 10% Paper Awards of IEEE MMSP 2013, and the Young Investigator Award of VCIP 2005. He was the recipient of the Outstanding Electrical Professor Award presented by Chinese Institute of Electrical Engineering in 2019, and Young Investigator Award presented by Ministry of Science and Technology, Taiwan, in 2006. He is currently the Chair of the Steering Committee of IEEE ICME. He has been serving as the President of the Chinese Image Processing and Pattern Recognition Association, Taiwan, since 2019. He has served as a Technical Program Co-Chair for IEEE ICME 2010, and a General Co-Chair for IEEE VCIP 2018, and a Technical Program Co-Chair for IEEE ICIP 2019. He has served as an Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE MULTIMEDIA, and *Journal of Visual Communication and Image Representation*.

Shin'ichi Satoh (Member, IEEE) received the B.E. degree in electronics engineering and the M.E. and Ph.D. degrees in information engineering from the University of Tokyo, Tokyo, Japan, in 1987, 1989, and 1992, respectively. He has been a Full Professor with the National Institute of Informatics, Tokyo, Japan, since 2004. He was a Visiting Scientist with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, from 1995 to 1997. His current research interests include image processing, video content analysis, and multimedia databases.