Deep Region Adaptive Denoising for Texture Enhancement

SEONG-EUI LEE1, SUNG-MIN WOO2, JONG-HAN KIM1, JE-HO RYU1, AND JONG-OK KIM1, (Member, IEEE)

1School of Electrical Engineering, Korea University, Seoul 02841, South Korea
2School of Electrical, Electronics and Communication Engineering, Korea University of Technology and Education, Cheonan 31253, South Korea

Corresponding authors: Jong-Ok Kim (jokim@korea.ac.kr) and Sung-Min Woo (innosm@koreatech.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (MSIT), Korean Government, under Grant 2020R1A4A4079705; and in part by the Education and Research Promotion Program of Korea University of Technology and Education (KoreaTech), in 2021.

ABSTRACT Image denoising is a highly challenging problem yet important task in image processing. Recently, many CNN-based denoising methods have made great performances but they commonly denoise blindly texture and non-texture regions together. This frequently leads to excessive texture smoothing and detail loss. To address this issue, we propose a novel region adaptive denoising network that adjusts the denoising strength according to region textureness. The proposed network conducts denoising tasks for texture and non-texture areas independently to improve the visual quality of the resulting image. To this end, we first generate a texture map that separates the image into texture and non-texture region. Because the difference between texture and non-texture is more evident in the frequency domain than in the spatial domain, the classification is performed through discrete cosine transform (DCT). Second, guided by the texture map, denoising is performed independently in two subnets, corresponding to the texture and non-texture regions. This allows the texture subnet to avoid excessive smoothing of high frequency details, and the non-texture subnet to maximize noise reduction in flat regions. Finally, a cross fusion that takes into intra- and inter-relationship between two resulting features is proposed. The cross fusion highlights the discriminant features from two subnets without degradation when combining the output of two subnets, and thus helps enhancing the performance of regions adaptive denoising. The superiority of the proposed method is validated on both synthetic and real-world images. We demonstrate that our method outperforms the existing methods in both objective scores and subjective image quality, in particular showing outstanding results in the restoration of visually sensitive textures. Furthermore, ablation study shows that our network can adaptively control the noise removal strength by manually manipulating the texture map and that the details of the texture region can be further improved. This also can simplify the cumbersome noise tuning process when deploying deep neural networks (DNN) architectures into products.

INDEX TERMS Region adaptive image denoising, convolution neural network, texture classification, cross attention, texture enhancement.

I. INTRODUCTION Image denoising is a fundamental and indispensable task in image processing. The common denoising approach reconstructs an original image by separating noises from its degraded version, typically modelled by $y = x + n$. Since the noise term $n$ and the original image term $x$ are not known, it is an ill-posed problem that makes it highly challenging and requires strong priors. Recently, CNN models [1], [2], [3], [4], [5], [6], [7] achieve a great denoising performances by learning strong priors from large scale data. Although many CNN-based denoising methods have made great performances, they still tend to excessively smooth high frequency components of texture regions. Basically, texture region has
a larger amount of high frequency components rather than non-texture region. However, most of the existing methods
commonly denoise blindly texture and non-texture regions

Texture map $M_t$

Texture subnet

Texture output $(t_i)$

($l_n$ $- l_t$)

Fusion

Output $(l_f)$

Non-texture map $M_{nt}$

Non-texture subnet

Non-texture output $(l_{nt})$

Non-texture region

FIGURE 1. Generic schematic of a region adaptive denoising based on a
texture map. The network adaptively controls the extent of noise removal
strength depending on the regions by two different denoising subnets.

Finally, the denoised features from two subnets are com-

bined using the cross-fusion module. Existing methods
for combining features commonly utilize concatenation or
element-wise multiplication of multiple features [8], [9].

However, they may result in the loss of the important fea-
tures to make a decisive contribution to the classification or
regression. Sometimes, they suffer from an unbalanced and
biased fusion toward one distinct output. To combine fea-
tures more effectively, [10], [11], [12] proposed the attention
module and its variants which consist of spatial and channel

attention blocks to improve the fusion performance. How-
ever, these methods still cannot effectively take the advantage
of the correlation between different features. The proposed
cross-fusion module permutes the results of the self and cross
attention to allow inter and intra feature fusion. This enables
us to effectively learn the correlation from the output features
of two subnets and preserve discriminative information for
adaptive denoising.

Furthermore, the proposed method can control the degree
of noise removal by simply adjusting the texture map with-
out an iterative training process to visually find the opti-

mal parameters. This is a great advantage when applied in
real-world use cases. Data-driven neural network approaches
are difficult to tune in, and different providers have different
optimal criteria for imaging systems. Unlike existing denoi-
sing methods, in the proposed system, the texture map not only
divides the image into textured and non-textured regions, but
also controls the strength of denoising in textured regions.
Therefore, it is easy to apply different levels of denoising by
simply replacing the texture map. A detailed experiment is
given in Section IV-D. In summary, our contributions are as
follows:

- A region adaptive denoising network is proposed to
  independently perform denoising tasks for texture and
  non-texture regions using two subnets.
- A texture map based on DCT classification is proposed to
  indicate the texture probability of the region, which
  enables the two denoising subnets to exclusively focus
  on their specific domain. The texture map is versatile
  for controlling the noise removal strength of a region,
  making it easy to find optimal tuning for deployment.
- A cross-fusion module is proposed to effectively
  fuse outputs of both subnets by exploiting intra- and
  inter-relationship between the textured and non-textured
  features.
- We conduct comprehensive denoising experiments and
  ablation studies to demonstrate the superiority of the
  proposed over existing methods.

We review related works in a section II and introduce our
proposed network in a section III. The experiment results
and analysis are provided in a section IV. Ablation studies are
provided in a section V. Finally, we discuss the conclusions
in a section VI.

II. RELATED WORKS

Recently, as deep learning models have achieved great per-
formance in denoising, many CNN-based denoising net-
works are emerging. Jain et al. [3] firstly proposed to apply
correlation neural network (CNN) for the denoising task.
Burger et al. [13] achieved a higher performance than BM3D using a plain multi-layer perceptron (MLP). Based on success of CNN architecture, Zang et al. [1] demonstrated residual learning and batch normalization make the training of deeper CNNs faster and more reliable. To further improve residual learning, [14] proposed DC-ResBlock to obtain a large receptive field. [15], [16], [17] proposed diverse methods to incorporate non-local operations in DNN for image denoising. Although these methods have exploited both local and non-local correlations, it was still challenging to fully exploit both correlations. To overcome this problem, [18] performed sparsity modeling for local and non-local correlations separately, which lead to optimal sparsity, and thus improved the denoising performance. Also, [19] proposed a noise basis learning method through subspace projection to maintain a non-local structure of input. However, these methods assume that the noise is spatially invariant. To deal with spatially variant noises, [2] proposed FFDNet to utilize a non-uniform noise level map. Inspired by FFDNet, [20] proposed deep plug-and-play IR to handle a wide range of noise levels by taking advantages of Resnet and UNet architecture. [21] used a hybrid orthogonal projection and estimation framework to improve the generalization capability of the network in the context of spatially varying and unknown noises. TSDNET [22] proposed a multi-scale attention module which utilizes both spatial and frequency features to reconstruct dehazed images. The heterogeneous features contribute to better performance.

Recently, as transformer model has shown great performance in natural language tasks [23], diverse denoising methods have adapted transformer-based model instead of CNN-based model [24], [25]. Vision transformer [26] divides image into a local patches and provides the sequence of local patches as an input to learn their correlation. Inspired by transformer architecture, [24] proposed a pre-trained model IPT for low-level vision task based on the transformer. However, since the computational complexity of self-attention in transformer increases quadratically with the size of the image, it has a limitation in denoising task using high-resolution image as an input. To overcome this problem, [25] applies self-attention across channel dimensions instead of spatial dimension to have linear complexity rather than quadratic.

Although CNN- and transformer-based denoising methods have shown tremendous success, they still suffer from preserving weak textures or high-frequency details. Generally noise and texture are very similar to each other in that both consist of high frequency. In other words, it is significantly challenging to distinguish noise from texture. On the other hand, the non-texture region composed of low-frequency components can be easily identified from noise. However, regardless of different frequency components depending on the regions, many denoising methods perform image denoising blindly with the designed loss aiming to restore the original pixel level on average, and thus, make it difficult to preserve edge and texture detail without degradation in the process. Therefore, denoising process frequently tends to smooth texture details as well as noises. This makes it difficult to keep original texture information during denoising. To overcome this problem, texture-preserving denoising methods such as Gradnet [27] and DCEF [28] have been proposed. Their common approach is to exploit edge map extracted by an edge detector in deep networks. However, these methods just feed the edge map into the network as a prior to preserve edge during denoising. Therefore, the map itself does not significantly affect the denoising task.

Unlike the previous works above-mentioned, the texture map in the proposed method represents the textureness score or probability for all regions of the image. This map divides the extracted features by regions and sends them to individual subnets for adaptive denoising. As a similar approach, we previously proposed NRIBP [29] which addresses to super-resolve a noisy image with texture-adaptive back-projection, but it is a non-deep-learning method for image super resolution. There is no work to adaptively perform deep learning based denoising depending on textureness.

Attention in neural networks is widely used to concentrate on important local regions and allows network to extract discriminative features. SENet [30] proposed channel attention to improve the representational capacity of network in image classification. CBAM [31] used dual attention modules that consist of channel and spatial sub-modules to better super-resolve image. It has been shown great performance in many applications, including image classification [32], action recognition [33], segmentation [34], [35], image captioning [36], [37], etc. Recently, attention methods are also used for diverse feature fusion tasks [10], [12], [38], [39], [40]. In [12], the attention method was used for fusing low dynamic images to a high dynamic range images. [10] proposed the cross attention network to combines contextual features and spatial
information. [39] proposed a HSI-LIDAR fusion based on cross attention, where the attention is used to highlight the inter-modal features between LiDAR and HSI. Based on the success of transformers in computer vision, various methods utilized transformer units to execute attention mechanism. Wei et al. [41] designed a transformer based cross-attention model for image and sentence matching by jointly exploiting the intra- and inter-modal relationships of image regions and sentence words. [42] proposed a dual-branch transformer to fuse image patches of different scales for image classification. Motivated by these studies, we propose a CF module that effectively combines texture and non-texture features. The proposed method extracts noise features that are distinguished from texture and non-texture regions in separate self-attention modules, and incorporates cross-attention to fuse inter-relationship between the texture and non-texture features.

III. THE PROPOSED METHOD

Existing denoising methods optimize a loss function over the entire image region without identifying texture and non-texture areas with different frequency distributions. This makes smooth textures and leaving background noises. The proposed method solves this problem by independently denoising from two subnets and fusing them with the novel combining method. The texture map adaptively steers the denoising process.

A. MOTIVATION FOR THE USE OF TEXTURE MAP

In general, texture and non-texture regions have different spectral distributions. Specifically, the texture has a larger amount of high-frequency rather than the non-texture. Noise is typically characterized by high frequency distribution. If it is mixed with texture region, it is considerably difficult to separate noises from the textures, which have certain overlap with noise in the frequency domain. This makes texture denoising highly challenging. On the other hand, noises in the non-texture region can be easily identified. Thus, both regions should be denoised adjustably according to the spectral distribution of signals. Especially, in the texture region, we need to compromise the extent of noise removal to preserve original high-frequency components. However, if we blindly estimate noise through one loss function without separating texture and non-texture, the signals of all regions are trained to be similar to ground-truth with an average value, hence essentially leaving background noise and smooth texture. To overcome this problem, we propose a deep learning based network that independently carries out denoising tasks for texture and non-texture and can adjustably control the extent of removal strength through texture map. Prior to the detailed illustration of the our proposed network, we firstly verify that whether texture and non-texture have different spectral distributions as follows:

We compare the high frequency magnitude between texture and non-texture. To obtain patches for texture and non-texture, we used sobel edge detection to extract the gradient components of images in Div2K dataset [43], then defined $100 \times 100$ regions with the highest gradient value for each image as texture, and conversely, $100 \times 100$ regions with the lowest gradient value as non-texture. To analyze the high-frequency components of each local region, we built the texture and non-texture local patches by randomly cropping images with $8 \times 8$ patches. Fig. 2 plots the high-frequency distribution acquired by DCT for 3000 pairs of texture and non texture local patches. It can be observed that high frequency distributions are clearly different for both texture and non-texture regions. This observation is effectively exploited for texture classification.
B. THE GENERATION OF GROUND-TRUTH TEXTURE MAP
In this paper, we first classify texture regions probabilistically in the image, and the classification output is named by texture map. To obtain a texture map, texture is classified with a deep network as shown in Fig. 3 (a). The classification is done on DCT domain because the difference between texture and non-texture is more evident on frequency domain rather than spatial one. An image is partitioned into $8 \times 8$ patches, and each patch is transformed through DCT. Note that $8 \times 8$ patch was found experimentally to be quite effective for texture classification. The transformed patch is put into the texture classification network, which is learned in a supervised manner. We built a dataset which has 3000 pairs of texture and non-texture patches with texture labels. The network is trained to generate a texture probability value between 0 and 1 through the sigmoid activation function for texture and non-texture patches, respectively.

The trained network is used to generate a ground-truth texture map of the target image as illustrated in Fig. 3 (b). We first extract patches of size $8 \times 8$ from the target image, and put them into the texture classification network. The texture probability value from the network output is equally mapped to the local window $8 \times 8$ patch (denoted $P_i$), and embedded into the texture map. For estimating the texturiness of a pixel with its surrounding pixels, we calculate the average of the map values overlapped by $P_i$, and it is defined as the texturiness probability of each pixel. We can see in Fig. 3 (b) that as expected, the region of sun flowers is classified to texture (close to 1) while the regions of sky and leaves are to non-texture (close to 0).

C. REGION ADAPTIVE IMAGE DENOISING
With the texture maps obtained in the previous subsection, the proposed denoising network first generates a texture map for a noisy input, and then probabilistically decomposes the input image into texture and non-texture components based on this map. They are independently denoised through separate subnets, and the outputs of both subnets are combined to obtain a denoised image eventually. As illustrated in Fig. 4, the proposed region adaptive denoising network consists of four stages: the map generation module, texture and non-texture denoising subnets, the cross-fusion module and the reconstruction module.

1) MAP GENERATION MODULE
The map-generation module of the proposed network generates a clean texture map from the noise input image as shown in Fig. 4. The detailed description is as follows: First, we generate the ground-truth texture map as illustrated in Fig. 3 (b) by putting the ground-truth image into the texture classification network. Second, the map-generation module in Fig. 4 is trained to generate the texture map from the noise image using the ground-truth texture map. Let $I_n \in \mathbb{R}^{H \times W \times C}$ be a noise image, $M_t \in \mathbb{R}^{H \times W}$ a texture map and $G : \mathbb{R}^{H \times W \times C} \to \mathbb{R}^{H \times W}$ is a map generation process. The map generation module can be written as:

$$M_t = G(I_n)$$  \hspace{1cm} (1)

The module is trained by optimizing the loss which is the prediction error of the texture map, and it is expressed by

$$L_g = \| M_{gt} - G(I_n) \|_2$$  \hspace{1cm} (2)

where $M_{gt}$ is the ground-truth texture map obtained by DCT based texture classification in Fig. 3.
2) TEXTURE AND NON-TEXTURE DENOISING SUBNETS

As illustrated in Fig. 4, we first extract the features of the noisy input through the feature extraction layer composed of a 3 × 3 convolution layer, \( F_m \in \mathbb{R}^{H \times W \times K} \). By multiplying the input features with the resulting texture map, the texture feature is partitioned as

\[
F_t = F_m \odot M_t
\]

where \( \odot \) denotes the element-wise multiplication. Similarly, we obtain a non-texture map \( M_{nt} \) by inverting the texture map, and non-texture features are separated by

\[
F_{nt} = F_m \odot M_{nt}
\]

Then, the both separated features are fed into the non-texture subnet, \( D_{nt} \) and the texture subnet, \( D_t \). And the outputs of each subnet are expressed by

\[
I_t = D_t(F_t)
\]

\[
I_{nt} = D_{nt}(F_{nt})
\]

where \( I_{nt} \) and \( I_t \) denote the output of non-texture and texture subnet, respectively, and they describe the estimated residual (noise) of the proposed subnets. To adaptively control the intensity of denoising depending on the textureness of each region, the losses of both subnets are given by

\[
L_t = \| [M_t \odot (I_n - I_{nt})] - I_t \|_2^2
\]

\[
L_{nt} = \| [M_{nt} \odot (I_n - I_{nt})] - I_{nt} \|_2^2
\]

3) CROSS FUSION (CF) MODULE

\( D_t \) and \( D_{nt} \) extract the relevant noise information, \( I_t \) and \( I_{nt} \) from each subnet. \( I_t \) lacks high frequency components for \( D_t \) to preserve the texture as much as possible. \( I_{nt} \) extracted from \( D_{nt} \) has a large portion of high frequency corresponding to noise to smooth flat areas. To effectively fuse \( I_t \) and \( I_{nt} \) with inherently different noise characteristics, the CF module consisting of self- and cross attention is utilized to consider the intra- and inter-relationship, which is motivated by [25]. The two self-attention blocks, as illustrated in Fig. 5, enhance the region of interest within each intermediate feature, \( \hat{F}_t \) and \( \hat{F}_{nt} \). In contrast, the cross-attention relates \( \hat{F}_t \) and \( \hat{F}_{nt} \) to obtain mutually exchangeable information on the context of texture-aware denoising. Note that \( \hat{F}_t \) and \( \hat{F}_{nt} \) are \( \mathbb{R}^{N \times K} \).

Self attention module consists of two sub-layers, multi-head attention layer and feed-forward layer as shown in Fig. 5. In multi-head attention layers, the attention is obtained by projecting the queries, keys, and values. Specifically, given a layer normalized tensor \( X \in \mathbb{R}^{H \times W \times C} \), we calculate query, key, and value for the input:

\[
Q = W^Q X, \ K = W^K X, \ V = W^V X
\]

where \( Q \in \mathbb{R}^{HW \times C/n}, K \in \mathbb{R}^{C/n \times HW}, V \in \mathbb{R}^{HW \times C/n}, n \) is the number of heads and \( W^{(*)} \) means convolution layers, which consist of \( 1 \times 1 \) point-wise convolution and \( 3 \times 3 \) depth-wise convolution [25]. Next, we generate the attention map with dot product of key and query, and the attention function is performed as follows:

\[
\text{Attention}(Q, K, V) = V \cdot \text{softmax}(K \cdot Q / \alpha)
\]

where \( \alpha \) is a learnable parameter that decides the weight of attention map. After that, we concatenate the multi-head attention maps together, and transform features using feed forward layers for improving the representation learning.

Similar with the self attention module, the cross attention module involves the transformer unit as shown in Fig. 5 (b). Although the self attention module extracts the discriminative information from each domain feature, inter-relationship (e.g., the relationship between texture and non-texture features) is not allowed. To achieve a robust multi-domain fusion, two different domain feature maps have to be learned for fusion by sharing information about each domain feature. Therefore, we propose a ‘cross-attention’ mechanism.
to extract inter-relationship. The detailed formulation as follows: First, we feed two feature maps in separate multi-head attention layers, and generate query, key, value, given by
\[
Q_X = W_Q X, K_X = W_K X, V_X = W_V X
\] (11)
\[
Q_Y = W_Q Y, K_Y = W_K Y, V_Y = W_V Y
\] (12)
where X and Y are texture and non-texture feature maps. Next, attention map is generated by dot product of key and query from different domains (texture and non-texture), and the attention function is carried out as
\[
\text{Attention}(Q_Y, K_X, V_X) = V_X \cdot \text{softmax}(K_X \cdot Q_Y / \alpha)
\] (13)
\[
\text{Attention}(Q_X, K_Y, V_Y) = V_Y \cdot \text{softmax}(K_Y \cdot Q_X / \alpha)
\] (14)
By doing this, feature maps can learn correlation from different domain features. Then updated feature maps are fed into the feed forward layers, and finally we can obtain a inter-domain relationship by fusing these feature maps through convolution layer.

An overall architecture of our proposed CF module is shown in Fig. 5 (a). In order to effectively reconstruct the final outputs by fusing both texture and non-texture features, we concatenate self- and cross-attention outputs, and finally, feed them into a self-attention module. Therefore, we can synthetically exploit intra-relationship within each domain and inter-relationship between texture and non-texture features through CF module.

### 4) RECONSTRUCTION FOR FINAL OUTPUT
The aggregated multi-channel feature map, \( \hat{F} \) from the CF module is converted to the noise output \( \hat{I}_{fn} \) by the reconstruction layers as follows.
\[
\hat{I}_{fn} = R(\hat{F})
\] (15)
where \( R \) represents the reconstruction layer. Finally, the denoised image is obtained by \( \hat{I}_f = I_n - \hat{I}_{fn} \). The loss for the reconstruction module is given by
\[
L_f = \|I_{gt} - \hat{I}_f\|_1
\] (16)

### D. DETAILED NETWORK ARCHITECTURE AND LOSS FUNCTION
Our texture classification network consists of two 3 × 3 convolution layers and three fully-connected layers as shown in Fig. 3 (a). Each convolution layer is followed by ReLU activation function and max pooling layer. The network is trained to generate a texture probability value between 0 and 1 through the sigmoid activation function for texture and non-texture patches, respectively.

As illustrated in Fig. 4, our proposed network consists of a map generation subnet (\( G \)) and two denoising subnets (\( D_{nt}, D_t \)), and these subnets have the same structure based on UNet [54]. Each subnet has four encoder and decoder stages with skip connections, where feature maps are \( \frac{1}{2} \) scale downsampled with a 4 × 4-stride-2 convolution at each last encoder stage and 2 scale up-samples with 2 × 2 deconvolution before each decoder stage. Skip connections pass low-level feature maps which contain detailed raw information from each encoder stage to its corresponding decoder stage. The basic convolution blocks in map-generation subnet and two denoising subnets follow the same residual convolution architecture which is composed of two 3 × 3 convolution

---

**TABLE 1.** Comparison with the denoising approaches on three synthetic datasets with AWGN.

| Datasets | CBM3D [44] | WNNM [45] | NCSR [46] | DnCNN [11] | MemNet [4] | PPDNet [2] | UDNet [15] | VDN [47] | Gradnet [27] | NBnet [19] | Proposed |
|----------|------------|-----------|-----------|------------|----------|-----------|-----------|---------|------------|----------|---------|
| \( \sigma = 15 \) | LIVE1 | 32.85 | 31.70 | 31.46 | 35.72 | 33.84 | 33.96 | 33.74 | 33.94 | 34.15 | 34.25 | 34.34 |
| Set5 | 33.42 | 32.92 | 32.57 | 34.04 | 34.18 | 34.30 | 34.19 | 34.34 | 34.51 | 34.64 | 34.71 |
| BSD68 | 32.67 | 31.27 | 30.84 | 33.87 | 33.76 | 33.85 | 33.76 | 33.90 | 34.07 | 34.15 | 34.18 |
| \( \sigma = 25 \) | LIVE1 | 30.05 | 29.15 | 29.05 | 31.23 | 31.26 | 31.37 | 31.09 | 31.50 | 31.54 | 31.73 | 31.79 |
| Set5 | 30.92 | 30.61 | 30.33 | 31.88 | 31.98 | 32.10 | 31.82 | 32.24 | 32.30 | 32.51 | 32.54 |
| BSD68 | 29.83 | 28.62 | 28.31 | 31.22 | 31.17 | 31.21 | 31.02 | 31.35 | 31.39 | 31.54 | 31.54 |
| \( \sigma = 50 \) | LIVE1 | 26.98 | 26.07 | 26.06 | 27.95 | 27.99 | 28.10 | 27.82 | 28.36 | 28.30 | 28.55 | 28.61 |
| Set5 | 28.16 | 27.58 | 27.20 | 28.95 | 29.10 | 29.25 | 28.87 | 29.47 | 29.39 | 29.70 | 29.70 |
| BSD68 | 26.81 | 25.86 | 25.73 | 27.91 | 27.91 | 27.95 | 27.76 | 28.19 | 28.12 | 28.35 | 28.41 |

**TABLE 2.** Comparison with the denoising approaches on three synthetic datasets with AWGN.

| Datasets | FFDNet [2] | DSNet [48] | GradNet [27] | RPCNN [49] | BRDNet [50] | RNAN [51] | RDN [52] | IPT [24] | DRNet [14] | MWDCNN [53] | Proposed |
|----------|------------|-----------|-----------|------------|----------|-----------|---------|------|----------|----------|---------|
| \( \sigma = 15 \) | LIVE1 | 33.87 | 33.91 | 34.07 | 34.10 | - | - | - | - | - | 34.18 |
| Set5 | 34.63 | 34.63 | 34.85 | - | 34.88 | - | - | - | - | - | 35.17 |
| McMuster | 34.66 | 34.67 | 34.81 | - | 35.08 | - | - | - | - | - | 35.16 |
| BSD68 | 34.68 | 34.68 | 34.85 | - | 35.08 | - | - | - | - | - | 35.18 |
| \( \sigma = 25 \) | LIVE1 | 31.21 | 31.28 | 31.39 | 31.24 | 31.43 | - | - | - | - | 31.47 | 31.45 |
| Set5 | 32.13 | 32.16 | 32.35 | 32.43 | 32.41 | - | - | - | - | - | 32.70 |
| McMuster | 32.35 | 32.40 | 32.45 | 32.33 | 32.75 | - | - | - | - | - | 32.83 |
| BSD68 | 32.91 | 32.91 | 32.95 | 32.93 | 32.93 | 32.93 | 32.93 | 32.93 | 32.93 | 32.93 | 32.93 |
| \( \sigma = 50 \) | LIVE1 | 27.96 | 28.05 | 28.12 | 28.06 | 28.16 | 28.27 | 28.31 | 28.39 | 28.20 | 28.13 | 28.41 |
| Set5 | 28.48 | 29.05 | 29.23 | 29.23 | 29.23 | 29.58 | 29.56 | 29.64 | 29.26 | 29.26 | 29.67 |
| McMuster | 29.18 | 29.28 | 29.39 | 29.33 | 29.52 | 29.72 | - | - | - | - | 29.80 |
layers, each followed by LeakyReLU activation function. The reconstruction layer consists of one encoder and decoder stage, and each stage consists of the residual convolution block.

To classify the regions of the image and remove noises according to the characteristics of regions in an end-to-end manner, we design the overall objective of our network as,

$$L = L_f + \alpha(L_t + L_m) + \beta L_g$$

where $\alpha$ and $\beta$ denote the hyperparameters for the denoising subnet loss and the map generation loss. $\alpha = 0.2$ and $\beta = 0.01$ were found to be experimentally appropriate and applied to the entire paper unless otherwise specified.

IV. EXPERIMENT

A. TRAINING SETTINGS

Throughout the study, the DIV2K [43] and Color BSD (CBSD) [56] datasets are used to evaluate the denoising performance of synthetic noise images. DIV2K consists of 1000 natural images that are approximately 2000 x 1400 pixels in size. The CBSD dataset is part of the Berkely Segmentation Dataset and Benchmark, consisting of 500 images. Among them, 800 training images for DIV2K and 432 training images for CBSD are used in the training process. Also,
we used 200 validation images for DIV2K in the validation process. Adam optimizer [57] is used with the momentums of (0.9, 0.999), and the image are randomly cropped to 128 x 128 patches for the input of the network. The learning rate is initialized as $10^{-4}$ and decays to $2 \times 10^{-1}$ every 500 epochs.

In this paper, we evaluate the proposed network with the noise images that are corrupted by spatially invariant AWGN. To evaluate our proposed network for Gaussian denoising, we synthetically generate training dataset [43], [56] with the different Gaussian noise levels ($\sigma = 15, 25, 50$) and train our network for each noise level.

B. DENOISING PERFORMANCE

To quantitatively evaluate the denoising performance of the proposed method, we use five datasets: BSD68 [56], Set5 [61], LIVE1 [55], kodak24 [59], McMaster [58] with the different Gaussian noise levels ($\sigma = 15, 25, 50$). Table 1 summarizes the PSNR results of each Gaussian noise level compared to the CNN-based denoising methods. The proposed method outperforms all compared methods. GradNet utilizes a gradient map to better preserve texture details, and thus can be directly compared to the proposed method. However, the proposed method achieves an average 0.22 dB higher PSNR than GradNet. Compared to the recently proposed NBNet, our method shows better results at the sigma (15, 25), but it is similar or slightly higher when noise variance is 50². The proposed method not only reconstructs a clean image, but also trains the internal subnet to handle texture separately. Even in the case of heavy noises, where high frequency signals are dominant, as illustrated in Figs. 6 and 7, the proposed method tries to reconstruct the original texture.

FIGURE 8. Denoising examples from McMaster [58]. Our results superior in weak texture regions such as lines and dots.

FIGURE 9. Denoising examples from Kodak24 [59]. The image restoration quality of our results is more faithful to the ground-truth than other results.
S.-E. Lee et al.: Deep Region Adaptive Denoising for Texture Enhancement

FIGURE 10. Denoising examples from real-world noisy images [60]. Our results show better denoising performance in preserving and reconstructing texture details in the real-world dataset as well as in synthetic datasets.

FIGURE 11. Denoising result depending on the non-texture map manipulation. The meaning of $M_{nt}(x)$ indicates that the non-texture map is uniformly filled with $x$. The result shows that texture denoising subnet prevents high-frequency details and non-texture subnet maximizes noise reduction, hence our network can adaptively adjust the strength of noise removal through the non-texture map.

from the messy and unrecognizable image using the texture map. Therefore, when noise becomes strong ($\sigma = 50$), PSNR is slightly degraded due to the unique property that prioritizes the texture reconstruction. Even though the quantitative PSNR improvement of the proposed method is negligible compared to NBNet when noise variance is $50^2$, subjective visual quality is compelling, especially on texture regions as illustrated in Figs. 6 and 7. It can be observed that our results show superiority in preserving the weak-texture details than the existing methods.

Table 2 illustrates a quantitative comparison between the previous methods. Especially, we compare with the transformer-based model IPT. Compared to the existing methods, our proposed method reports competitive results on quantitative evaluation. Even though IPT achieves higher PSNR than our proposed method for the noise level 50 on McMaster dataset, subjective visual quality of our method is superior to others in preserving the fine-scale textured regions and high-frequency components as illustrated in Fig. 8. Also, as shown in Fig. 9, our method shows better performance than other approaches in restoring the head and features of a person, which are important texture detail of the image. It is worth noting that, while other methods focus on only quantitative evaluation criteria such as PSNR, our proposed method improves quantitative and subjective image quality performances simultaneously by allowing texture region and non-texture region to be processed independently.

C. DENOISING REAL-WORLD NOISY IMAGES

In a real world, noise arises not only from the sensor, but also from the various image processing steps of the camera pipeline, such as gamma correction, quantization, white balancing, JPEG compression, etc. Therefore, it is difficult to model the noise distribution in a uniform or Gaussian manner. In order to evaluate the practicality of the proposed method in practical use cases, we exploit RNI 15 dataset [60], which contains 15 real noise images. Since the proposed method is a non-blind denoising model, we approximated the noise level as $\sigma = 50$ for the evaluation, and compared with existing
methods. Because RNI 15 dataset has no ground truth images, we illustrate some of resulting images in Fig. 10. It can be seen that, unlike the other methods smooth texture details and noise simultaneously, the proposed method preserves more textures and structures, similar to the evaluation results of synthetic data.

D. BOOSTING TEXTURE REGION WITH ADJUSTING NON-TEXTURE MAP

In general, CNN-based denoising methods conducts blind denoising, which finds optimal parameters by learning strong priors from large scale data. Thus, if we need to adjust the intensity of noise removal, these methods should control the denoising strength by conducting iterative training process. Unlike these methods, our network can simply control the degree of noise removal by adjusting the non-texture map. To be specific, our non-texture map acts as an indicator that informs the textureness of the input image to the network, hence our proposed network can control the denoising strength with the probability value of the non-texture map.

To prove this, we evaluate the non-texture maps manipulated with arbitrary values as illustrated in Fig. 11. When non-texture map is set to 1, all regions are assumed to be non-texture and all regions are strongly denoised, leading to over-smoothing as shown in Fig. 11 (a). However, when non-texture map is filled with 0 (i.e., all regions are assumed to be texture), the strength of denoising is relatively weak, and texture details are kept as shown in Fig. 11 (c). If non-texture map is uniformly 0.5, this means that the region is neither texture nor non-texture. So, it is denoised with medium...
FIGURE 13. Visual comparison with and without non-texture map. Note that the non-texture map plays a decisive role in reconstructing the fine pattern of the Buddha image by allowing texture and non-texture to be processed independently.

FIGURE 14. Comparison of concatenation and the cross fusion.

FIGURE 15. Comparison between (a) gray and (b) binary scale non-texture map.

TABLE 3. Ablation study of our proposed architecture.

| Method                                              | PSNR(dB)  |
|-----------------------------------------------------|-----------|
| DnCNN (single network + w/o map)                    | 31.23dB   |
| DnCNN (individual subnets + w/ map + concat)        | 31.60dB   |
| DnCNN (individual subnets + w/ map + cross fusion)  | 31.65dB   |
| Proposed (single network + w/o map)                 | 31.55dB   |
| Proposed (individual subnets + w/ map + concat)     | 31.71dB   |
| Proposed (individual subnets + w/ map + cross fusion)| 31.79dB   |

strength as shown in Fig. 11 (b). It can be seen that texture denoising subnet preserves high-frequency details and non-texture subnet maximizes noise reduction, hence our network can adaptively adjust the strength of noise removal through the non-texture map. Therefore, if the subtle adjustments in denoising level are required, this can be done by simply adjusting the non-texture map, without redundant process of preparing noisy images with different target sigma and training the entire network from scratch.

Fig. 12 presents qualitative comparisons of texture regions by adjusting the non-texture map. And also we compare the subjective visual quality with the Restormer. In order to demonstrate the efficiency of our network, we just change the non-texture map without additional update of the network for other results. Fig. 12 (a) is the output of our proposed network with the original non-texture map from learning. Figs. 12 (b) and (c) are the results that multiply parameters (0.8 and 0.2) for increasing the texture probability value (close to 0) of the texture regions. Especially, to prevent the non-texture regions from being affected, we set a threshold value of the non-texture map and only multiplied parameters in the texture regions. As shown in Fig. 12 (b), with the increase of texture probability value, texture details are kept better and SSIM is also improved while PSNR is slightly degraded compared to Fig. 12 (a). In addition, it is also observed that further improvement of texture probability makes the texture detail more distinct as shown in Fig. 12 (c). Furthermore, as illustrated in Fig. 12 (d), our results show significantly improved texture quality over Restormer by simply adjusting the non-texture map. It can be seen that as the texture probability increases, the high-frequency components
are better preserved. Although there is a slight reduction in PSNR, the detail of the texture region is clearly improved. Therefore, our proposed network can control the noise removal strength by adjusting the non-texture map and the detail of the texture region can be further improved as needed with some quantitative compromise.

V. ABSTRACT STUDY
To demonstrate the effectiveness of our proposed network architecture, we evaluate the two main ideas of our model. First, we examine the effectiveness of two-branch network structure using texture map. Second, an ablation study is performed to find out how the proposed cross-fusion module affects the overall reconstruction performance.

A. EFFECT OF TWO DENOISING SUBNET STRUCTURE
To validate how individual subnet architecture and non-texture map are involved in improving denoising performance, we design a baseline without non-texture map (i.e. a single denoising subnet without non-texture map) and compare it with the proposed architecture as illustrated in Fig. 13. The baseline without the non-texture map excessively smooths the texture region such as the forehead and hair as shown in Fig. 13 (d), while the proposed subnet architecture reconstructs the subtle patterns of the Buddha’s surface by preventing the textures from being smooth in Fig. 13 (e).

In the quantitative comparison as shown in Table 3, the use of individual subnets results in a performance improvement of more than 0.16 dB over the baseline without a non-texture map. It can be seen that by separating the region of image (texture and non-texture) through non-texture map, our proposed network can adjust the denoising strength according to the characteristics of the area and enhance the denoising performance, especially on the texture regions.

B. CROSS FUSION MODULE
As described in Section III.C.3, we proposed the CF module that extracts discriminant information and simultaneously...
learns inter-relationships between texture and non-texture features. In order to evaluate the effectiveness of our CF module, we compare the proposed CF with simple concatenation. Fig. 14 (d) illustrates the result images by substituting the CF with the concatenation. Compared to the proposed CF in Fig. 14 (e), the performance is degraded especially in reconstructing texture patterns. Furthermore, the proposed method with the CF module (individual subnets + w/ map + cross fusion) achieves about 0.08 dB higher than just concatenation for fusion (individual subnets + w/ map + cross fusion) as shown in Table 3.

C. SUBSTITUTING $D_t$ AND $D_{nt}$ WITH DnCNN
As with the previous subsections, we analyze the effect of the proposed network architecture to use DnCNN as baseline. As shown in Table 3, DnCNN (individual subnets + w/ map + concat) achieves about 0.37dB higher than DnCNN and DnCNN (individual subnets + w/ map + cross fusion) achieves about 0.42dB higher than DnCNN. It can be seen that our proposed network architecture universally achieves a great performance using any other networks as well as our proposed network.

D. GRAY VS. BINARY SCALE NON-TEXTURE MAP
The significance of the non-texture map is examined by comparing a binary scale map with a gray one. The binary scale map means that all pixels are either texture or non-texture, while the textureness of each pixel is probabilistically expressed in the gray scale map. As shown in Fig. 15 (b), using a binary map results in excessive smoothing on weak texture regions such as the line shape which has less textureness because those regions are classified as non-texture. On the other hand, those weak texture regions are enhanced better for a gray scale map. We can see that denoising strength can be controlled elaborately for all regions with a finer non-texture map.

E. NOISE SENSITIVITY ANALYSIS
Most denoising methods work on the assumption that the noise level was perfectly predicted. However, in reality, the real noise level is neither known nor estimated perfectly. Therefore, noise mismatch is essentially occurred when removing the real noise. If the estimated noise level is higher than the real noise level (over-estimated), image details are removed with noise. On the other hand (under-estimated), the noise is not completely removed. Therefore, we evaluate in case of noise mismatch to assess the noise sensitivity of our proposed network in comparison with FFDNet, DnCNN, DRNet, IPT. Since the noise level map is not used in our proposed network such as BM3D and FFDNET, we used the model trained on $\sigma = 50$ for evaluating on BSD68 images with noise level ranging from 0 to 60. The noise level sensitivity curves are shown in Fig. 17. Our method shows better results in both cases of underestimating and overestimating the real noise level. Especially, our network results are superior when underestimating the real noise level.

Qualitative results for noise sensitivity in underestimating are shown in Fig. 16. The methods trained on $\sigma = 50$ were evaluated on BSD68 images with different gaussian noise levels ($\sigma = 15,35$). The second row is the result of each model for noise image ($\sigma = 15$) and third row is the result of each model for noise image ($\sigma = 35$). In the case of overestimation, the existing methods remove noise from the entire area at a certain intensity, leading to smoothing texture regions. However, since the proposed method extracts the texture map from the noise image, it is able to adjust the noise removal strength based on the texturenness of regions. Therefore, our results show superior in preserving the texture details compared to the existing methods.

VI. CONCLUSION
In this paper, we provide a novel denoising method that adaptively controls the extent of noise removal strength according to image regions. We first propose a classification network to divide an image into texture and non-texture region, which works on DCT domain, and generate a texture map. Using this texture map, we propose the region adaptive denoising network that conducts independently denoising tasks for texture and non-texture regions. By fusing the denoised results for both regions, we adjust the denoising strength according to the texture characteristics of regions and further enhanced texture denoising performance by minimizing texture smoothness. Various experiments have demonstrated that the proposed network can control the noise removal strength using texture maps and achieve better performance than blind denoising.

REFERENCES
[1] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” IEEE Trans. Image Process., vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
[2] K. Zhang, W. Zuo, and L. Zhang, “FFDNet: Toward a fast and flexible solution for CNN-based image denoising,” IEEE Trans. Image Process., vol. 27, no. 9, pp. 4608–4622, Sep. 2018.
[3] V. Jain and S. Seung, “Natural image denoising with convolutional networks,” in Proc. Adv. Neural Inf. Process. Syst., vol. 21, 2008, pp. 1–15.
[4] Y. Tai, J. Yang, X. Liu, and C. Xu, “MemNet: A persistent memory network for image restoration,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 4539–4547.
[5] M. Gharbi, G. Chaurasia, S. Paris, and F. Durand, “Deep joint denoising and denoising,” ACM Trans. Graph., vol. 35, no. 6, pp. 1–12, Nov. 2016.
[6] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, “CycleISP: Real image restoration via improved data synthesis,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 2696–2705.
[7] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, “Toward convolutional blind denoising of real photographs,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 1712–1722.
[8] L. Gómez-Chova, D. Tuia, G. Moser, and G. Camps-Valls, “Multimodal classification of remote sensing images: A review and future directions,” Proc. IEEE, vol. 103, no. 9, pp. 1560–1584, Aug. 2015.
[9] D. Rao, M. De Deuge, N. Nourani–Vatani, S. B. Williams, and O. Pizarro, “Multimodal learning and inference from visual and remotely sensed data,” Int. J. Robot. Res., vol. 36, no. 1, pp. 24–43, Jan. 2017.
[10] M. Liu and H. Yin, “Cross attention network for semantic segmentation,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Sep. 2019, pp. 2434–2438.
[11] X. Zhang and X. Wang, “MARN: Multi-scale attention retinex network for low-light image enhancement,” IEEE Access, vol. 9, pp. 50939–50948, 2021.
[12] J.-H. Kim, J.-H. Ryu, and J.-O. Kim, “FDD-MEF: Feature-decomposition-based deep multi-exposure fusion,” IEEE Access, vol. 9, pp. 164551–164561, 2021.

[13] H. C. Burger, C. J. Schuler, and S. Harmeling, “Image denoising: Can plain neural networks compete with BM3D?” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2012, pp. 2392–2399.

[14] J. Zhang, Y. Zhu, W. Li, W. Fu, and L. Cao, “DRNet: A deep neural network with multi-layer residual blocks improves image denoising,” IEEE Access, vol. 9, pp. 79936–79946, 2021.

[15] S. Lefkimmiatis, “Universal denoising networks: A novel CNN architecture for image denoising,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3204–3213.

[16] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang, “Non-local recurrent network for image restoration,” in Proc. Adv. Neural Inf. Process. Syst., vol. 31, 2018, pp. 1–10.

[17] S. Lefkimmiatis, “Non-local color image denoising with convolutional neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 3587–3596.

[18] H. Liu, J. Zhang, and R. Xiong, “CAS: Correlation adaptive sparse modeling for image denoising,” IEEE Trans. Comput. Imag., vol. 7, pp. 638–647, 2021.

[19] S. Cheng, Y. Wang, H. Huang, D. Liu, H. Fan, and S. Liu, “NNNet: Noise basis learning for image denoising with subspace projection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 4896–4906.

[20] K. Zhang, Y. W. Zuo, L. Zhang, L. Van Gool, and R. Timofte, “Plug-and-play image restoration with deep denoiser prior,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 10, pp. 6360–6376, Oct. 2022.

[21] S. A. Khowaja, B. N. Yahya, and S.-L. Lee, “Cascaded and recursive ConvNets (CRCNN): An effective and flexible approach for image denoising,” Signal Process., Image Commun., vol. 99, Nov. 2021, Art. no. 116420.

[22] R. W. Liu, Y. Guo, Y. Lu, K. T. Chui, and B. B. Gupta, “Deep network-enabled haze visibility enhancement for visual IoT-driven intelligent transportation systems,” IEEE Trans. Ind. Informat., early access, Apr. 27, 2022, doi: 10.1109/TII.2022.3170594.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 1–15.

[24] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, X. Wu, and W. Gao, “Pre-trained image processing transformer,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 12299–12310.

[25] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, “Restorer: Efficient transformer for high-resolution image restoration,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 5728–5739.

[26] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, A. F. Goyal, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, T. G晗, M. Rupp, J. Uszkoreit, and N. Houlsby, “An image is worth 16 words: Trans- formers for image recognition at scale,” 2020, arXiv:2010.11929.

[27] Y. Liu, S. Anwar, L. Zheng, and Q. Tian, “GradNet image denoising,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2020, pp. 508–509.

[28] S. Chupraphawan and C. A. Ratanamahatana, “Deep convolutional neural network with edge feature for image denoising,” in Proc. Int. Conf. Comput. Inf. Technol. Cham, Switzerland: Springer, 2019, pp. 169–179.

[29] J.-S. Yoo and J.-O. Kim, “Noise-robust iterative back-projection,” IEEE Trans. Image Process., vol. 29, pp. 1219–1232, 2020.

[30] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7132–7141.

[31] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional block attention module,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 3–19.

[32] M. F. Stollenga, J. Masci, F. Gomez, and J. Schmidhuber, “Deep networks with internal selective attention through feedback connections,” in Proc. Adv. Neural Inf. Process. Syst., vol. 27, 2014, pp. 1–9.

[33] Y. Fan, S. Weng, Y. Zhang, B. Shi, and Y. Zhang, “Context-aware cross-attention for skeleton-based human action recognition,” IEEE Access, vol. 8, pp. 15290–15293, 2020.

[34] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, “Dual attention network for scene segmentation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 3146–3154.
[60] M. Lebrun, M. Colom, and J.-M. Morel, “The noise clinic: A blind image denoising algorithm,” *Image Process. Line*, vol. 5, pp. 1–54, Jan. 2015.

[61] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. A. Morel, “Low-complexity single-image super-resolution based on nonnegative neighbor embedding,” in *Proc. Brit. Mach. Vis. Conf.* BMVA Press, 2012, pp. 135.1–135.10.

SEONG-EUI LEE received the B.S. degree in electrical engineering from Dongguk University, Seoul, South Korea, in 2021. He is currently pursuing the M.S. degree in electrical engineering with Korea University, Seoul. His current research interests include deep learning-based various image processing and computer vision algorithm.

SUNG-MIN WOO received the B.S. degree in electrical engineering from Stony Brook University, Stony Brook, NY, USA, in 2006, the M.S. degree from the Pohang University of Science and Technology, South Korea, in 2008, and the Ph.D. degree from Korea University, South Korea, in 2020. From 2008 to 2020, he participated in research and development on mobile camera systems at the LG Electronics Mobile Communication Division. He is currently working as an Assistant Professor with the School of Electrical, Electronics and Communication Engineering, Korea University of Technology and Education. His current research interests include color constancy, image and video processing, computer vision, and machine learning.

JONG-HAN KIM received the B.S. degree in electronic engineering from Kwangwoon University, Seoul, South Korea, in 2020. He is currently pursuing the M.S. degree with Korea University, Seoul. His current research interests include deep learning-based various image processing and computer vision algorithms.

JE-HO RYU received the B.S. degree in electrical engineering from Korea University, Seoul, South Korea, where he is currently pursuing the integrated M.S. and Ph.D. degrees in electrical engineering. His research interests include machine learning and deep learning in particular with applications on computer vision tasks, such as visual recognition and image enhancement tasks, including high dynamic range imaging, image fusion, and super resolution.

JONG-OK KIM (Member, IEEE) received the B.S. and M.S. degrees in electronic engineering from Korea University, Seoul, South Korea, in 1994 and 2000, respectively, and the Ph.D. degree in information networking from Osaka University, Osaka, Japan, in 2006. From 1995 to 1998, he worked as an Officer with Korea Air Force. From 2000 to 2003, he was with the SK Telecom Research and Development Center and Mubeworks Inc., South Korea, where he was involved in research and development on mobile multimedia systems. From 2006 to 2009, he was a Researcher with the Advanced Telecommunication Research Institute International (ATR), Kyoto, Japan. He joined Korea University, in 2009, where he is currently a Professor. His current research interests include image processing, computer vision, and intelligent media systems. He was a recipient of the Japanese Government Scholarship, from 2003 to 2006.

***