USIM Gate: UpSampling Module for Segmenting Precise Boundaries concerning Entropy

Kyungsu Lee
DGIST

Haeyun Lee
Samsung SDI

Jae Youn Hwang
DGIST

Abstract

Deep learning (DL) techniques for precise semantic segmentation have remained a challenge because of the vague boundaries of target objects caused by the low resolution of images. Despite the improved segmentation performance using up/downsampling operations in early DL models, conventional operators cannot fully preserve spatial information and thus generate vague boundaries of target objects. Therefore, for the precise segmentation of target objects in many domains, this paper presents two novel operators: (1) upsampling interpolation method (USIM), an operator that upsamples input feature maps and combines feature maps into one while preserving the spatial information of both inputs, and (2) USIM gate (UG), an advanced USIM operator with boundary-attention mechanisms. We designed our experiments using aerial images where the boundaries critically influence the results. Furthermore, we verified the feasibility that our approach effectively segments target objects using the Cityscapes dataset. The experimental results demonstrate that using the USIM and UG with state-of-the-art DL models can improve the segmentation performance with clear boundaries of target objects (Intersection over Union: +6.9%; Boundary Jaccard: +10.1%). Furthermore, mathematical proofs verify that the USIM and UG contribute to the handling of spatial information.

1 Introduction

Deep learning (DL) models have shown remarkable performance in the extraction of contextual features from synthetic images. In the early era, many DL models have been studied to invent novel operators for effective feature extraction in segmentation tasks. A fully convolutional network (FCN) deploys skip operators to compensate for spatial information (Long et al., 2015). The U-Net architecture includes convolutions in the contracting path, transposed convolutions for upsampling in the expansive path, and skip connections to compensate for locality (Ronneberger et al., 2015). FusionNet is built on the autoencoder architecture, including residual blocks that effectively increase the depth of the network for a better optimization (Quan et al., 2016). Moreover, advanced DL models have been developed based on vanilla networks. For instance, U-NetPPL was designed based on the U-Net model with pyramid pooling layers (Kim et al., 2018). In addition, based on the U-Net architecture, AU-Net was developed with attention gates, which are substitute for skip connections and generate attention-weighted feature maps (Oktay et al., 2018). Recently, modern DL models have been developed to improve segmentation performance. DeepLab V3+ introduces multiple operators, such as atrous separable convolution, spatial pyramid pooling, and depth-wise separable convolution for a better semantic segmentation task (Chen et al., 2018). Moreover, several studies have applied the attention mechanism, initially designed for language processing, to DL models and improved segmentation performance. Dual attention networks (DANets) deploy a dual attention module with a self-attention mechanism to enhance feature representations (Fu et al., 2019). In addition, criss-cross network (CCNet) verifies the state-of-the-art (SoTA) performance in segmentation tasks using a novel criss-cross attention module that precisely captures contextual information (Huang et al., 2019).

Moreover, further studies have been conducted to enhance the boundaries of target objects in images for a better segmentation. The efficient sub-pixel convolutional neural network (ESPCN) was introduced with the *pixel shuffling* method, which includes a sub-pixel convolution layer for a super-resolution that improves the peak signal-to-noise ratio and results in clear boundaries of segmented objects in images (Shi et al., 2016). The device

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206. Copyright 2023 by the author(s).

	Attention Mechanism	Boundary Enhancement	Spatial Information	SoTA
FCN (Long et al., 2015)				✗
U-Net (Ronneberger et al., 2015)			✓	✗
U-NetPPL (Kim et al., 2018)			✓	✗
FusionNet (Quan et al., 2016)			✓	✗
AU-Net (Oktay et al., 2018)	✓			
DeepLab V3+ (DLV3+) (Chen et al., 2018)		✓	✓	
DANet (Fu et al., 2019)	✓		✓	✓
CCNet (Huang et al., 2019)	✓		✓	✓
ESPCN (Shi et al., 2016)		✓		
DLR (Marmanis et al., 2018)		✓		
RPCNet (Zhen et al., 2020)		✓		✓
TreeUNet (Yue et al., 2019)			✓	
Red-Net (Hua et al., 2019)			✓	
CA-Conv-BiLSTM (CCB) (Liu and Ji, 2020)	✓			✓
Ours	✓	✓	✓	✓

Table 1: Summary of key features of various segmentation models related to our approach. ✓marker in SotA indicates the state-of-the-art models published within 3 years (2019-2021), whereas ✗marker in SotA indicates vanilla network.

level ring (DLR) network was developed with a remarkably improved multi-scale boundary prediction by *ensemble learning*, which utilizes boundary probability maps (Marmanis et al., 2018). Furthermore, the remote procedure call network (RPCNet) achieves the precise boundary segmentation of target objects using a spatial gradient fusion that suppresses non-semantic edges (Zhen et al., 2020). Although these methods enhance the boundaries of target objects in synthetic images, the precise semantic segmentation of multiple objects in aerial images is still challenging owing to the lack of boundary details and spatial information.

Therefore, several architectures customized for aerial images have been proposed for improved segmentation. TreeUNet was designed based on an adaptive network using customized blocks that calculate the lower triangular matrix (Yue et al., 2019). The class-wise attention-based convolutional and bidirectional long short-term memory (LSTM) network utilizes a feature extraction module and an attention mechanism with LSTM modules for the segmentation tasks of aerial imagery (Hua et al., 2019). RedNet was developed to utilize a wide range of depth inferences with a recurrent encoder–decoder structure (Liu and Ji, 2020). However, despite the precise localization of target objects by DL models, aerial image segmentation remains challenging for the following reasons (Maggiori et al., 2017a): (1) low resolution of images impeding boundary or edge detection, (2) spatial information loss hampering precise boundary segmentation, and (3) trade-off between detection and localization by pooling layers and convolution striding during the feature extraction process.

In this study, to precisely segment target objects with accurate boundaries, we designed two novel operators: upsampling interpolation method (USIM) and USIM gate (UG). The USIM is an upsampling and merging operator that allows DL models to significantly improve boundary-oriented segmentation performance using the preserved spatial information of inputs and increased information entropy. In addition, we applied the attention gate mechanism (Oktay et al., 2018) to the USIM, which is denoted as "UG." The UG is a boundary-attention module that generates feature maps, including boundary-related attention coefficients extracted from input features, which are extracted by the USIM. We applied our operation in multiple fields (aerial image, Cityscapes image (Cordts et al., 2016)), and we show an experimental analysis of the aerial image with a significant improvement of boundaries among them.

In summary, the main contributions of this paper are as follows[1]:

- We developed two operators: (1) **USIM**, which is an upsampling and merging operator with the preservation of pixel values and spatial information and increased information entropy, and (2) **UG**, which is an attention mechanism-based operator highlighting the boundary-oriented area of target objects.
- We mathematically and experimentally proved the strengths of implementing the USIM and UG in any SoTA model.
- The best-performing model with the SoTA model and UG outperformed any segmentation models owing to its precise segmentation of multiple objects with accurate boundaries in aerial images (Intersection over Union (IoU): +6.9%; Boundary Jaccard (BJ): +10.1%).

[1]Our code is available at https://github.com/kyungsu-lee-ksl/USIM-GATE

2 Method

This section illustrates the detailed descriptions of the USIM and USIM Gate (UG) operators for DL models.

2.1 Modeling

To illustrate the definition and advantages of USIM and UG, we utilize the symbols for the following operators; USIM ($\circledast$), Addition ($\oplus$), Concatenation ($\odot$), Hadamard Product (element-wise product, $\otimes$), Matrix Product ($\circ$ or omitted), Activation (σ), and Pooling ($\mathscr{P}$). In addition, the name of the operator with italic form can be utilized as the function e.g. $F^{(i)} \circledast F^{(j)} = USIM(F^{(i)}, F^{(j)})$.

Let M be a CNN model, and Θ be the set of parameters in M. Here, DL models consecutively produce the feature map using convolution operation. Hence, let set of feature maps generated by M with Θ in the segmentation task be $F(M;\Theta)$, and let the i^{th} feature maps in $F(M;\Theta)$ be $F^{(i)} \in F(M;\Theta)$. Then, $F^{(i)} \in \mathbb{R}^{H^{(i)} \times W^{(i)} \times C^{(i)}}$, where $H^{(i)}, W^{(i)}, C^{(i)}$ are height, width, and number of channels of $F^{(i)}$, respectively. Here, we denoted the spatial elements of $F^{(i)}$ as $f^{(i)}_{h,w}$ where $h = \{1,2,...,H^{(i)}\}$ and $w = \{1,2,...,W^{(i)}\}$ such that $f^{(i)}_{h,w} \in \mathbb{R}^{C^{(i)}}$, and $f^{(i)}_{h,w,c}$ where $c = \{1,2,...,C^{(i)}\}$ such that $f^{(i)}_{h,w,c} \in \mathbb{R}$. Furthermore, the USIM and UG are analyzed in terms of entropy, and the entropy of feature map ($F^{(i)}$) is denoted as $H : \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{R}$, such that $H(F^{(i)}) \in \mathbb{R}$ (Shannon, 2001).

2.2 Upsampling Interpolation Method (USIM)

The schematic description of the USIM is illustrated in Fig. 1, and the procedures of the USIM are described in Algorithm 1. In line 7, $[x]$ indicates a greatest integer function such that $[x] = max\{n \in \mathbb{Z}; n \leq x\}$.

USIM is designed to simultaneously include a merging and an upsampling operator to replace the operators such as concatenation and addition blocks. To this end, the USIM alternately arranges the pixels of two input feature maps into the output feature map (merge), which has twice the height and width of input feature maps (upsample). Therefore, the USIM using two input feature maps ($F^{(i)} \in \mathbb{R}^{H \times W \times C}$ and $F^{(j)} \in \mathbb{R}^{H \times W \times C}$) generates the feature map ($F^{(k)} = F^{(i)} \circledast F^{(j)} = USIM(F^{(i)}, F^{(j)})$), such

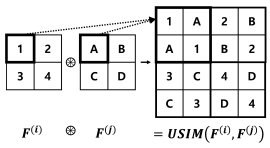


Figure 1: Schematic description of USIM which upscaling two feature maps ($F^{(i)}$ and $F^{(j)}$) and combining them into one.

Algorithm 1 USIM operation

Input: $F^{(i)}, F^{(j)} \in \mathbb{R}^{H \times W \times C}$

Output: $F^{(k)} \in \mathbb{R}^{2H \times 2W \times C}$

1: **Let** spatial element of $F^{(i)}$ and $F^{(j)}$ be $f^{(i)}_{h,w}$ and $f^{(j)}_{h,w}$
2: **And** $h = \{1,2,...,H\}$ and $w = \{1,2,...,W\}$
3: **Then** $f^{(i)}_{h,w}, f^{(j)}_{h,w} \in \mathbb{R}^C$
4: **Let** spatial element of $F^{(k)} \in \mathbb{R}^{2H \times 2W \times C}$ be $f^{(k)}_{h',w'}$
5: **And** $h' = \{1,2,...,2H\}$ and $w' = \{1,2,...,2W\}$
6: **Then** $f^{(k)}_{h',w'} \in \mathbb{R}^C$
7: **Then**

$$f^{(k)}_{h',w'} = \begin{cases} f^{(i)}_{[\frac{h'+1}{2}][\frac{w'+1}{2}]} & (\text{if } 0 \equiv h' + w' \mod 2) \\ f^{(j)}_{[\frac{h'+1}{2}][\frac{w'+1}{2}]} & (\text{if } 1 \equiv h' + w' \mod 2) \end{cases}$$

8: **Return** $F^{(k)}$

that $F^{(k)} \in \mathbb{R}^{2H \times 2W \times C}$. Here, the USIM is channel-wise operator and thus $f^{(i)}_{h,w}, f^{(j)}_{h,w}, f^{(k)}_{h,w} \in \mathbb{R}^C$. Note that, the USIM is a binary operation such as addition and multiplication, not a trainable operator like convolution. In addition, the following equation is used for the backpropagation:

$$\mathscr{P}_{\text{gap}}(f^{(k)}_{h,w}; 2) = 2(f^{(i)}_{h,w} + f^{(j)}_{h,w}) \tag{1}$$

, where $\mathscr{P}_{\text{avg}}(x; s)$ is global average pooling operation of x with the size of s. In addition, Equation (1) implies that the USIM is a linear transformation, and thus the USIM performs the same role as identical mapping (He et al., 2016). Therefore, the USIM as an identical mapping has the advantage that previously extracted features can be delivered as preserved and that the forward and backpropagated gradients are well transferred, improving the gradient vanishing problem.

2.3 USIM Gate

As illustrated in Fig. 2, using two feature maps ($F^{(i)}, F^{(j)} \in \mathbb{R}^{H \times W \times C}$), one output feature map is generated by the UG. In the upper stream, $F^{(i)}$ and $F^{(j)}$ are merged by concatenation ($F^{(i)} \odot F^{(j)} \in \mathbb{R}^{H \times W \times 2C}$), and the merged feature map is upsampled by the deconvolution, which is denoted as $deconv : \mathbb{R}^{H \times W \times 2C} \rightarrow \mathbb{R}^{2H \times 2W \times C}$. In contrast, in the lower stream, $F^{(i)}$ and $F^{(i)}$ are merged and upsampled by the USIM such that the feature map ($F^{(i)} \circledast F^{(j)} \in \mathbb{R}^{2H \times 2W \times C}$) is generated. Therefore, two feature maps($F^{(m)}$ and $F^{(n)}$)are generated by the deconvolution after concatenation and USIM operation as the following:

$$\begin{aligned} F^{(m)} &= deconv(F^{(i)} \odot F^{(j)}), \quad s.t.\ F^{(m)} \in \mathbb{R}^{2H \times 2W \times C} \\ F^{(n)} &= F^{(i)} \circledast F^{(j)}, \quad\quad\quad\quad s.t.\ F^{(n)} \in \mathbb{R}^{2H \times 2W \times C} \end{aligned} \tag{2}$$

Here, $F^{(m)}$ and $F^{(n)}$ are utilized as a gating vector and a target pixel vector as illustrated in the Attention Gate

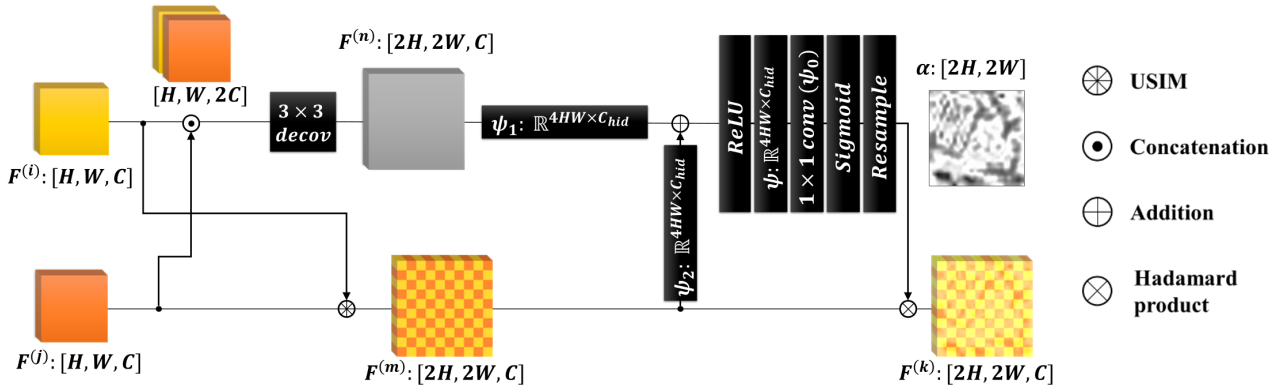


Figure 2: Pipeline of the USIM Gate (UG). The Attention Gate generates the output computed using the Hadamard product of spatial attention coefficient (α) and the feature map generated by USIM ($F^{(i)} \circledast F^{(j)}$).

in the AU-Net (Oktay et al., 2018). In addition, $F^{(m)}$ and $F^{(n)}$ are linearly reshaped using $flat : \mathbb{R}^{2H \times 2W \times C} \rightarrow \mathbb{R}^{(4HW) \times C}$ function. As shown in Fig. 2, the linear transformations and the activations (σ_1 and σ_2) are computed to $flat(F^{(m)})$ and $flat(F^{(n)})$ using the weights terms ($\psi_1, \psi_2, \psi \in \mathbb{R}^{(4HW) \times C_{hid}}$), and the bias terms ($b_1 \in \mathbb{R}^{C_{hid} \times C}$ and $b_2 \in \mathbb{R}^{4HW}$) as the follows:

$$F^{(l)} = \sigma_1(\psi_1^T flat(F^{(m)}) + \psi_2^T flat(F^{(n)}) + b_1) \tag{3}$$

$$\alpha = \sigma_2\big(\psi F^{(l)} \psi_0^T + b_2\big), \ s.t. \ \alpha \in \mathbb{R}^{4HW} \tag{4}$$

, where σ_1 and $\sigma 2$ are the ReLU and sigmoid activation, respectively, such that $\sigma_1(x) = max(0, x)$ and $\sigma_2(x) = \frac{1}{1+e^{-x}}$. Here, the sigmoid activation is used instead of the softmax activation since the consecutive usage of the softmax yields sparser activations. In addition, C_{hid} indicates number of hidden channels (16 in this paper), $\psi_0 \in \mathbb{R}^{1 \times C}$ is weight term indicating 1×1 convolution, and α indicates the spatial attention coefficient. The spatial attention coefficient (α) is resampled to the shape of $[2H, 2W]$, and the output of UG is the Hadamard product of spatial attention coefficient and the feature map generated by USIM. In summary, the UG is computed as follows:

$$F^{(k)} = UG(F^{(i)}, F^{(i)}) = \alpha \otimes (F^{(i)} \circledast F^{(j)}) \tag{5}$$

3 Theoretical Analysis

As a motivation for USIM, we pointed out entropy decreasing in upsampling. As illustrated in Fig. 3, the mechanism of USIM alternatively maps the pixel values of input feature maps one by one into a new feature map. Intuitively, we can consider two mechanisms of USIM; [Fig. 3 (a)] diagonally arranges the pixel values, or [Fig. 3 (b)] linearly arranges the pixel values. If the pixels are arranged as [A,A;1,1], in a second manner [Fig. 3 (a)], less pixel information and more duplicated values pass more often in some receptive fields. To avoid the issue, we arranged pixels as diagonal (See [A,A;1,1;B,B] and [A,1;1,A;B,2] in the blue box in Fig. 3). In addition, we conducted simple experiments. The results demonstrated that the USIM in a second manner exhibits the coarse boundary in the predicted segmentation maps, not as intended, and this is because of the frequent duplicated pixel values in the receptive fields, thus decreased entropy. Therefore, the USIM is designed in a first manner, as illustrated in Fig. 3 (a).

It is antecedently studied that if the entropy after convolution and merging operation is increased, better feature extraction can be possible (Biesiada et al., 2005). To demonstrate the increased entropy by using the USIM, Shannon entropy (Shannon, 2001) and pixel-level entropy (Wang et al., 2018) are used. The Shannon-Entropy of the feature maps($F^{(i)}$ and $F^{(j)}$), USIM, concatenation, and addition operator are formulated as $H(F^{(i)})$, $H(F^{(j)})$, $H(F^{(i)} \circledast F^{(j)})$, $H(F^{(i)} \odot F^{(j)})$, and $H(F^{(i)} \oplus F^{(j)})$, respectively. Likewise, the pixel-level entropy of feature maps, USIM with pooling operation, and addition operator are formulated as $H(f^{(i)}_{h,w,c})$, $H(f^{(j)}_{h,w,c})$, $H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c})))$, and $H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c})$, respectively. The convolution operation is denoted as *conv*.

Lemma 1. The pixel-level entropy and Shannon entropy of a merged image by USIM is compared to others after being resized as half ($\mathscr{P}$). That is, the pixel-level

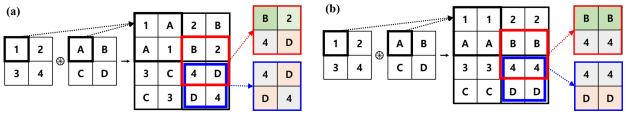


Figure 3: Mechanism of two versions of USIM

entropy of the original image $\left(H(max(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))\right)$, a merged image by an addition operation $\left(H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c})\right)$, and a pooled image after being merged by USIM $\left(H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c})))\right)$ are compared with the random variables ($f^{(i)}_{h,w,c}$ and $f^{(j)}_{h,w,c}$) of pixels at i and j of two images. In addition, Shannon entropy of the original image $\left(max(H(F^{(i)}), H(F^{(j)}))\right)$, Shannon entropy of a merged image by the addition operation ($H(F^{(i)} + F^{(j)})$), and Shannon entropy of a pooled image after being merged by USIM $\left(H(\mathscr{P}(USIM(F^{(i)}, F^{(j)})))\right)$ are compared with input images ($F^{(i)}$ and $F^{(j)}$).

Lemma 2. While the entropy of a merged image by USIM is without resizing, the merged image by the addition operation is resized as the same size as the merged image by USIM. That is, Shannon entropy of the original image ($max(H(F^{(i)}), H(F^{(j)}))$), a merged image by the addition operation ($H(intp(F^{(i)} + F^{(j)}))$), and a pooled image after being merged by USIM ($H(USIM(F^{(i)} + F^{(j)}))$) are compared with input images ($F^{(i)}$ and $F^{(j)}$) and a resizing operation.

Therefore, we can induce Theorem 1 from Lemma 1, Lemma 2.

Theorem 1. Shannon and pixel-level entropy of USIM are increased than those of the original feature maps and addition operator as the following:

- $H(F^{(i)} \circledast F^{(j)}) \geq max(H(F^{(i)}), H(F^{(j)}))$
- $H(F^{(i)} \circledast F^{(j)}) \geq H(F^{(i)} \oplus F^{(j)})$
- $H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))) \geq max(H(f^{(i)}_{h,w,c}), H(f^{(j)}_{h,w,c}))$
- $H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))) \geq H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c})$

Theorem 1 deals with entropy only in upsampling. For theoretically analyzing our approach, we try to verify the change in entropy when using the convolution operation.

Lemma 3. The pixel-level entropy of a resized image after being merged by USIM is compared to the pixel-level entropy of the merged image by others.

Lemma 4. While the merged image by USIM is without resizing, the merged image by the addition operation or the concatenation operation is resized as the same size as the merged image by USIM. The entropy of the generated images by the convolution operation after being merged by each operation are compared.

Therefore, we can induce Theorem 2 from Lemma 3, Lemma 4.

Theorem 2. Shannon entropy of USIM with convolution operator is increased than the that of a convolution after concatenation operator and a convolution after addition operator as the following:

- $H(conv(F^{(i)} \circledast F^{(j)})) \geq H(conv(F^{(i)} \oplus F^{(j)}))$
- $H(conv(F^{(i)} \circledast F^{(j)})) \geq H(conv(F^{(i)} \odot F^{(j)}))$

Theorems 1 and 2 imply that the generated feature map by the USIM operator contains more Shannon and pixel-level entropy than the feature maps generated by other operators. Therefore, it is concluded that the USIM can provide increased information and entropy for better feature extractions. The detailed mathematical proofs and the comparisons of entropy are illustrated in Appendix A.

4 Experimental Analysis

In the experiments, we compared the USIM and UG to other operators, and the performances of our best performing model with the UG were compared with those of other DL models. Table 1 lists the DL models used in the experiments. In this section, we only used the aerial image for experiments. A performance of the Cityscapes dataset is illustrated in Appendix B.

4.1 Dataset and Metrics

To evaluate the USIM, UG, and other DL models, we utilized four aerial image datasets of Inria (Maggiori et al., 2017b), WHU (Liu and Ji, 2020), Korean Urban Dataset (KUD) (Kim et al., 2018), and LoveDA (Wang et al., 2021). Here, Inria and WHU are binary building datasets, and they were utilized to validate the novel performance of the USIM and UG as compared to other operators. Meanwhile, KUD and LoveDA are multi-object segmentation datasets in aerial images, and they were utilized as benchmarks for the DL model with the UG and other SoTA models. In addition, to evaluate the segmentation performance, four metrics, namely, precision (prec.), recall, mean intersection over union (mIoU), and boundary Jaccard (BJ) (Fernandez-Moral et al., 2018), were mainly utilized. Because prec., recall, and mIoU are not sufficient to measure the fine segmentation performance, BJ, which measures the boundaries of objects, was utilized. Furthermore, the implementation details, training environment, description of datasets, and image distributions, including $k-$fold ($k = 10$) cross-validation, are illustrated in Appendix B.

4.2 Qualitative Results

Fig. 4 illustrates the qualitative results. The segmentation results by the best performing models in each group are illustrated, and ours indicates the TreeUNet with UG. The results illustrate that our model segments multi-class

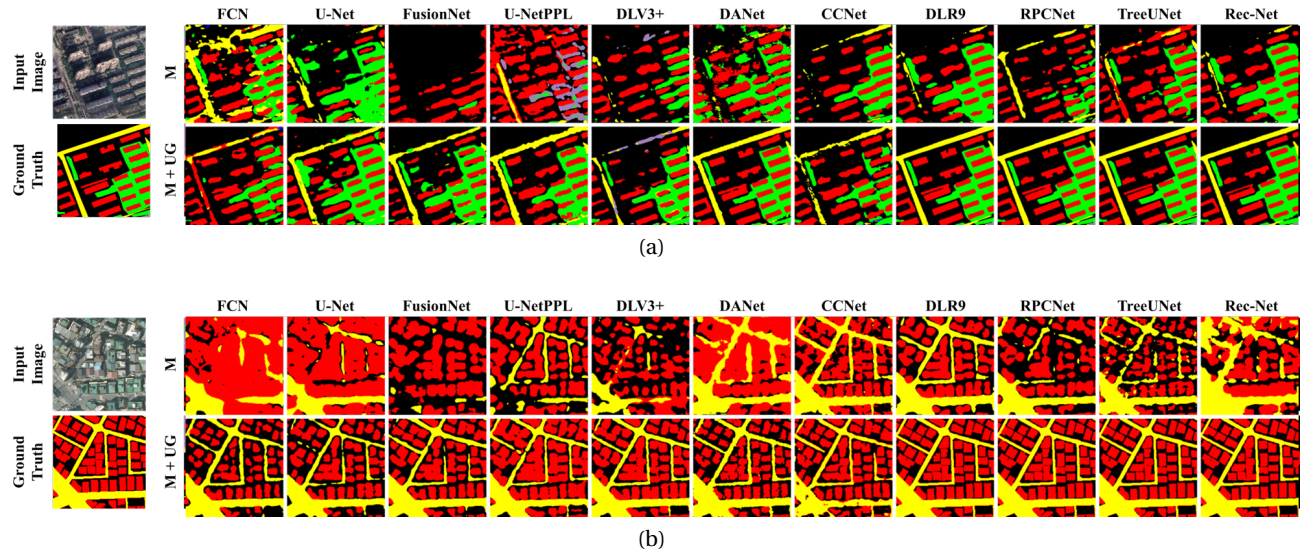


Figure 4: Qualitative results on each dataset. (a) Segmentation results on **LoveDA** dataset. (b) Segmentation results on **KUD** dataset.

objects with precise boundaries of the target objects, especially in buildings. The results imply the best model with the UG successively segments multi-class objects in aerial images.

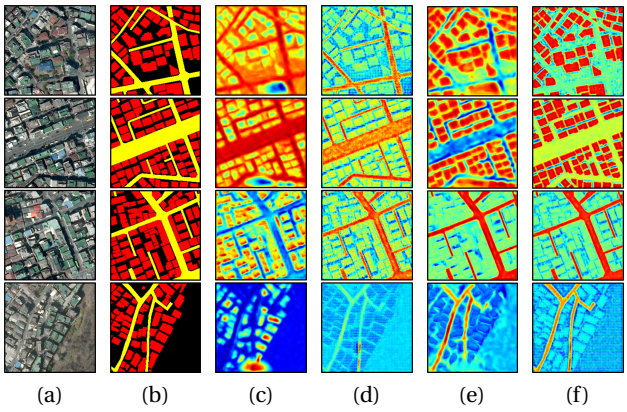


Figure 5: (a) Raw input image. (b) Ground truth for buildings (yellow) and roads (red). (c) Grad-CAM of 3rd-upsampling layer with deconvolution and concatenation (d) Grad-CAM of 3rd-upsampling layer with USIM, (e) Grad-CAM of 4th-upsampling layer with deconvolution and concatenation, and (f) Grad-CAM of 4th-upsampling layer with USIM. First two rows targeted building class and other two targeted roads class.

To verify the boundary-oriented segmentation performance of the UG, gradient-weighted class activation mapping (Grad-CAM) was utilized (Selvaraju et al., 2017). Fig. 12 illustrates the Grad-CAM outputs of upsampling layers in the U-Net with the USIM and concatenation using KUD datasets of building and road segmentation. Red indications high activation, whereas blue indicates low activation. The results illustrate that the Grad-CAM generated by upsampling layers in the USIM with the UG shows boundary-oriented attention distributions (red colors in buildings and roads) than concatenation after deconvolution. Therefore, the UG exhibits more explicit activations on the boundaries of the target objects than deconvolution and concatenation.

4.3 Quantitative Results

Because the boundaries are more critical for segmenting buildings than other objects in aerial images, the building segmentation datasets (Inria and WHU) were utilized to verify the outstanding performance and boundary-

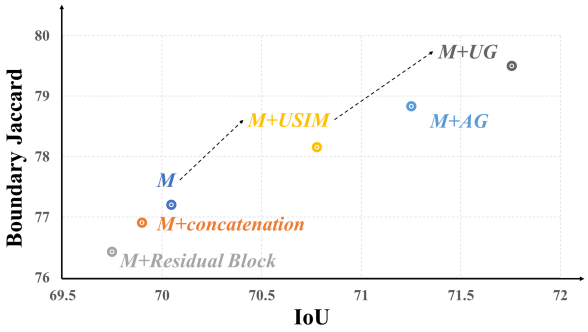


Figure 6: Average IoU and BJ score of the baseline model (*M*) with merging operators including USIM Gate (UG), Attention Gate (AG), USIM, concatenation, and residual block on Inria and WHU benchmark.

Model	Evaluations			Class-wise IoU (BJ)						
	prec.	recall	mIoU	bg	building	road	water	barren	forest	agriculture
AU-Net	58.52	75.42	45.6	48.4 (28.3)	34.3 (11.0)	45.1 (20.0)	44.5 (22.8)	51.7 (28.1)	39.8 (18.4)	55.8 (31.0)
CCB	60.71	77.2	48.2	50.9 (31.3)	39.3 (14.5)	47.8 (22.7)	46.7 (25.2)	53.2 (29.8)	41.8 (20.5)	58.0 (33.7)
CCNet	60.69	77.1	48.2	49.7 (30.0)	39.8 (14.8)	46.8 (21.7)	46.6 (25.2)	52.8 (29.4)	44.6 (23.1)	57.6 (33.2)
DANet	57.91	75.27	45.0	48.0 (28.0)	31.0 (9.1)	42.7 (17.4)	45.9 (24.3)	49.6 (25.5)	41.7 (20.2)	56.3 (31.5)
DLV3+	58.83	75.78	46.0	49.4 (29.5)	34.3 (11.0)	44.1 (18.9)	44.8 (23.1)	51.8 (28.1)	40.9 (19.6)	56.8 (32.2)
DLR9	59.02	75.82	46.2	49.8 (29.9)	34.7 (11.2)	43.7 (18.4)	44.9 (23.3)	51.3 (27.6)	42.9 (21.5)	56.8 (32.1)
ESPCN+M	61.79	78.21	49.6	52.2 (32.6)	40.4 (15.4)	47.5 (22.4)	48.3 (26.9)	54.3 (31.2)	44.7 (23.3)	59.8 (36.0)
Red-Net	60.67	77.19	48.1	52.4 (32.8)	37.9 (13.6)	46.3 (21.1)	45.4 (23.8)	52.9 (29.4)	44.0 (22.6)	58.1 (33.8)
RPCNet	61.14	77.66	48.8	50.1 (30.3)	40.4 (15.2)	47.6 (22.6)	46.8 (25.3)	54.0 (30.8)	44.4 (23.1)	58.6 (34.6)
TreeUNet	62.11	77.9	49.9	51.4 (31.9)	43.4 (17.4)	47.0 (21.9)	48.9 (27.5)	54.3 (31.2)	45.1 (23.7)	59.7 (36.1)
U-NetPPL	60.08	76.77	47.4	50.6 (31.0)	36.6 (12.7)	44.9 (19.5)	46.3 (24.7)	51.1 (27.3)	43.4 (21.9)	59.5 (35.8)
Ours	**65.2**	**80.61**	**53.7**	**56.5 (37.7)**	**48.0 (21.6)**	**51.3 (26.7)**	**49.9 (28.8)**	**58.6 (36.5)**	**48.8 (27.7)**	**63.5 (41.0)**

Table 2: Quantitative results on LoveDA dataset.

attention characteristics of the UG. Fig. 6 illustrates the quantitative analysis of the UG and USIM and comparison with other merging operators, namely, attention gate, concatenation, and residual block with the baseline models (M). To evaluate the basic performance, vanilla models, such as FCN, U-Net, U-NetPPL, and FusionNet, were used as M (Table 1). In addition, the BJ and IoU, which were evaluated in the Inria and WHU datasets, were averaged. The results demonstrate that the UG significantly improves the performance of the baseline model as compared with other operators.

Table 3 illustrates the quantitative analysis of the UG implemented into any DL model on the LoveDA dataset. When using the UG, the segmentation performances by the UG increased by at least 3.5% to a maximum of 7.0%. In addition, the TreeUNet+UG exhibited the best segmentation performance of every metric on every dataset.

Furthermore, the best-performing model with the UG was compared with other DL models in the multi-object segmentation datasets of LoveDA and KUD. In both datasets, as illustrated in Tables 2 and 4, the best performing model (ours), which was implemented using TreeUNet with the UG, exhibits SoTA performance as compared to any other DL models in all groups. The quantitative results show that the best model with the USIM achieved a 6.9% improved mIoU and a 10.1% BJ score compared to others.

Model	Baseline		UG		Improve.	
	mIoU	BJ	mIoU	BJ	mIoU	BJ
FCN	43.9	21.3	49.5	26.8	+5.6%	+5.5%
FusionNet	44.7	21.9	49.9	27.2	+5.2%	+5.3%
U-Net	46.1	23.2	49.9	27.2	+3.8%	+4.0%
U-NetPPL	47.5	24.7	51.1	28.5	+3.7%	+3.8%
DLV3+	46.0	23.2	50.0	27.3	+4.0%	+4.1%
DANet	45.0	22.3	50.0	27.3	+5.0%	+5.0%
CCNet	48.3	25.3	51.8	29.2	+3.5%	+3.9%
DLR9	46.3	23.4	53.3	30.9	**+7.0%**	**+7.5%**
Red-Net	48.1	25.3	53.7	31.3	+5.5%	+6.0%
RPCNet	48.8	26.0	53.1	30.6	+4.2%	+4.6%
TreeUNet	49.9	27.1	**53.8**	**31.4**	+3.8%	+4.3%

Table 3: Quantitative analysis on baselines and adopting UG.

Model	Class-wise IoU (BJ)			
	bg	building	road	water
AU-Net	74.5 (39.9)	59.6 (32.1)	53.4 (29.2)	60.7 (43.5)
CCB	76.8 (44.7)	62.5 (36.0)	56.4 (32.7)	60.6 (43.4)
CCNet	76.5 (43.3)	61.0 (34.0)	54.8 (31.4)	61.2 (44.1)
DANet	76.0 (42.6)	60.6 (33.4)	54.9 (31.3)	61.4 (44.3)
DLV3+	74.9 (41.0)	59.5 (31.8)	55.1 (31.3)	61.0 (44.0)
DLR9	76.5 (43.7)	61.7 (34.9)	57.0 (33.8)	61.4 (44.5)
ESPCN+M	78.2 (46.2)	62.7 (36.5)	56.9 (34.1)	63.2 (46.7)
Red-Net	77.9 (46.3)	63.3 (37.2)	56.9 (33.6)	62.5 (45.7)
RPCNet	77.4 (45.5)	62.0 (35.4)	56.3 (32.8)	64.3 (48.0)
TreeUNet	77.1 (44.9)	62.2 (35.7)	57.0 (33.7)	61.4 (44.4)
U-NetPPL	76.0 (42.9)	61.2 (34.2)	55.6 (31.8)	63.5 (47.1)
Ours	**80.8 (51.4)**	**66.4 (41.7)**	**61.9 (40.5)**	**66.7 (51.3)**

Table 4: Quantitative results on KUD dataset.

5 Conclusion

In this study, we mathematically and experimentally demonstrated the outstanding performance of the USIM and UG. The USIM provides a better segmentation performance while preserving and increasing spatial information, and the UG exhibits boundary-attention-based feature extraction. The experimental results demonstrate that the best model with the UG significantly enhanced the segmentation performance. In the quantitative analysis, TreeUNet with the UG achieved 6.9% and 10.1% improved IoU and BJ scores as compared to other DL models for multi-objects segmentation. The main contribution of this study is the development of novel operators that can be implemented in any DL model with enhanced segmentation performance. However, finding the best hyperparameters is needed to further enhance the segmentation performance of the UG and other SoTA models. This aspect will be one of our future works.

Acknowledgements

his work was partially supported by the Korea Medical Device Development Fund grant funded by the Korea government (the Ministry of Science and ICT, the Ministry of Trade, Industry and Energy, the Ministry of Health & Welfare, the Ministry of Food and Drug Safety) (Project Number: 1711174564, RS-2022-00141185). Additionally, this work was supported by the Technology Innovation Program(20014214, Development of an elderly-friendly

wearable smart healthcare system and service for real-time quantitative monitoring of urination and defecation disorders) funded By the Ministry of Trade, Industry & Energy(MOTIE, Korea).

References

Biesiada, J., Duch, W., Kachel, A., Maczka, K., and Palucha, S. (2005). Feature ranking methods based on information entropy with parzen windows. In *International conference on research in electrotechnology and applied informatics*, volume 1, page 1.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Fernandez-Moral, E., Martins, R., Wolf, D., and Rives, P. (2018). A new metric for evaluating semantic segmentation: leveraging global and contour accuracy. In *2018 ieee intelligent vehicles symposium (iv)*, pages 1051–1056. IEEE.

Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., and Lu, H. (2019). Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3146–3154.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer.

Hua, Y., Mou, L., and Zhu, X. X. (2019). Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional lstm network for multi-label aerial image classification. *ISPRS journal of photogrammetry and remote sensing*, 149:188–199.

Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., and Liu, W. (2019). Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 603–612.

Kim, J. H., Lee, H., Hong, S. J., Kim, S., Park, J., Hwang, J. Y., and Choi, J. P. (2018). Objects segmentation from high-resolution aerial images using u-net with pyramid pooling layers. *IEEE Geoscience and Remote Sensing Letters*, 16(1):115–119.

Liu, J. and Ji, S. (2020). A novel recurrent encoder-decoder structure for large-scale multi-view stereo reconstruction from an open aerial dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6050–6059.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.

Maggiori, E., Charpiat, G., Tarabalka, Y., and Alliez, P. (2017a). Recurrent neural networks to correct satellite image classification maps. *IEEE Transactions on Geoscience and Remote Sensing*, 55(9):4962–4971.

Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. (2017b). Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IEEE.

Marmanis, D., Schindler, K., Wegner, J. D., Galliani, S., Datcu, M., and Stilla, U. (2018). Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135:158–172.

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., et al. (2018). Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*.

Quan, T. M., Hildebrand, D. G., and Jeong, W.-K. (2016). Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics. *arXiv preprint arXiv:1612.05360*.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626.

Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1):3–55.

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

Wang, J., Zheng, Z., Ma, A., Lu, X., and Zhong, Y. (2021). Loveda: A remote sensing land-cover dataset for domain adaptation semantic segmentation.

Wang, P., Fu, H., and Zhang, K. (2018). A pixel-level entropy-weighted image fusion algorithm based on

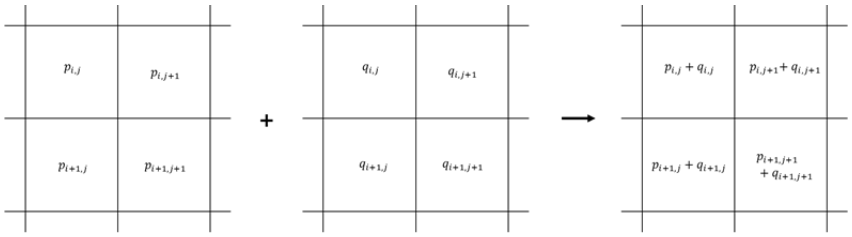
bidimensional ensemble empirical mode decomposition. *International Journal of Distributed Sensor Networks*, 14(12):1550147718818755.

Yue, K., Yang, L., Li, R., Hu, W., Zhang, F., and Li, W. (2019). Treeunet: Adaptive tree convolutional neural networks for subdecimeter aerial image segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 156:1–13.

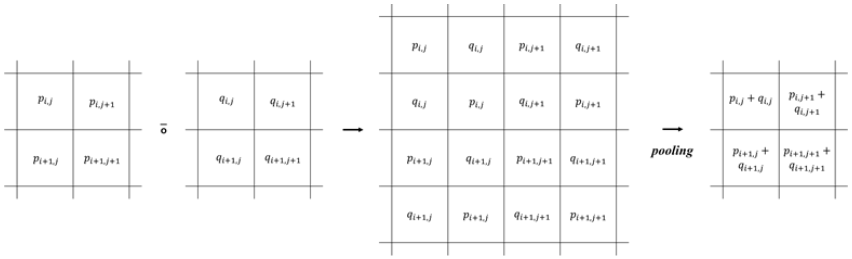
Zhen, M., Wang, J., Zhou, L., Li, S., Shen, T., Shang, J., Fang, T., and Quan, L. (2020). Joint semantic segmentation and boundary detection using iterative pyramid contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

A Appendix A. Mathematical proofs

Lemma 1. The pixel-level entropy and Shannon entropy of a merged image by USIM is compared to others after being resized as half ($\mathscr{P}$). That is, the pixel-level entropy of the original image $\left(H(max(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))\right)$, a merged image by an addition operation $\left(H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c})\right)$, and a pooled image after being merged by USIM $\left(H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c})))\right)$ are compared with the random variables ($f^{(i)}_{h,w,c}$ and $f^{(j)}_{h,w,c}$) of pixels at i and j of two images. In addition, Shannon entropy of the original image $\left(max(H(F^{(i)}), H(F^{(j)}))\right)$, Shannon entropy of a merged image by the addition operation ($H(F^{(i)} + F^{(j)})$), and Shannon entropy of a pooled image after being merged by USIM $\left(H(\mathscr{P}(USIM(F^{(i)}, F^{(j)})))\right)$ are compared with input images ($F^{(i)}$ and $F^{(j)}$).



(a) Addition operation



(b) USIM with pooling

Figure 7: The Schematic diagrams of (a) addition operation and (b) average pooling after USIM. The merged image that a pooling operation is applied to after being merged by USIM is the same as the merged image by the addition operation.

[Proof] First, the pixel-level entropy of the original images $\left(H(max(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))\right)$, a merged image by the addition operation $\left(H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c})\right)$, and a pooled image after being merged by USIM $\left(H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c})))\right)$ are compared. As shown in Figure 7, it is straightforward to see that a pooled image after being merged by USIM is totally the same as a merged image by the addition operation. Therefore, the entropy of the images by USIM and addition operations are totally same. That is, we have:

$$H(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))) = H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c}) \tag{6}$$

In addition, suppose that $Z = X+Y$ where, X and Y are the random variables that represent $f^{(i)}_{h,w,c}$ and $f^{(j)}_{h,w,c}$), respectively. With $P(Z = z|X = x) = P(Y = z - x|X = x)$, the following can be proven:

$$\begin{aligned} H(Z|X) &= \sum_x P(X = x) H(Z|X = x) \\ &= -\sum_x P(X = x) \cdot \sum_z P(Y = z - x|X = x) \log P(Y = z - x|X = x) \\ &= -\sum_x P(X = x) \sum_y P(Y = y|X = x) \log P(Y = y|X = x) \\ &= H(Y|X) \end{aligned} \tag{7}$$

Here, suppose X and Y are independent, then, $H(Y|X) = H(Y)$. With the mutual information $I(X;Z) \geq 0$, it is concluded

that $H(X+Y) = H(Z) \geq H(Z|X) = H(Y|X) = H(Y)$. Similarly, it can be shown that $H(X+Y) = H(Z) \geq H(Z|Y) = H(X|Y) = H(X)$. Thus, the following inequality holds:

$$H(X+Y) \geq max(H(X), H(Y)) \tag{8}$$

Therefore, in terms of the pixel-level entropy, the following inequality should hold as well:

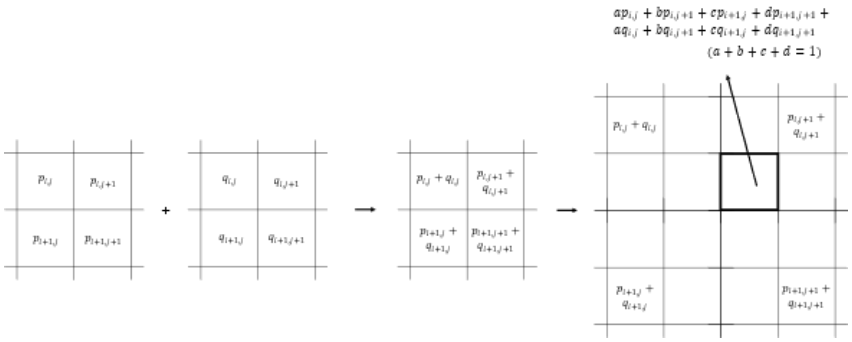
$$H(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c}) \geq max(H(f^{(i)}_{h,w,c}), H(f^{(j)}_{h,w,c})) \tag{9}$$

In conclusion, by combining (1) and (2), the inequality of the pixel-level entropy is given as:

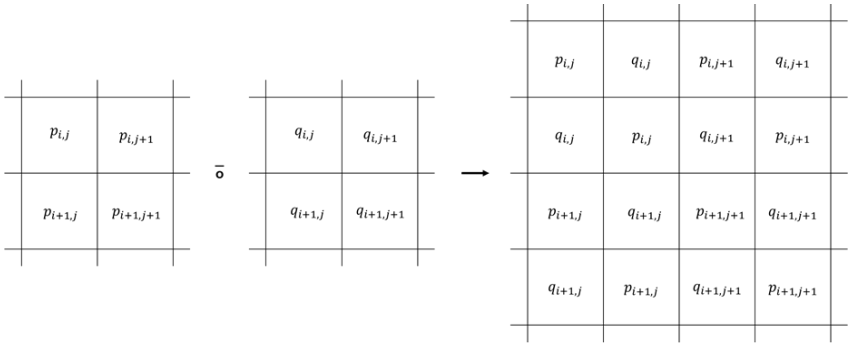
$$H\big(\mathscr{P}(USIM(f^{(i)}_{h,w,c}, f^{(j)}_{h,w,c}))\big) = H\big(f^{(i)}_{h,w,c} + f^{(j)}_{h,w,c}\big) \geq max(H(f^{(i)}_{h,w,c}), H(f^{(j)}_{h,w,c})) \tag{10}$$

The pixel-level information is increased when the addition and USIM operations are applied, and thus more informative feature extraction can be provided.

Lemma 2. While the entropy of a merged image by USIM is without resizing, the merged image by USIM is without resizing, the merged image by the addition operation is resized as the same size as the merged image by USIM. That is, Shannon entropy of the original image ($max(H(F^{(i)}), H(F^{(j)}))$), a merged image by the addition operation ($H(intp(F^{(i)} + F^{(j)}))$), and a pooled image after being merged by USIM ($H(USIM(F^{(i)} + F^{(j)}))$) are compared with input images ($F^{(i)}$ and $F^{(j)}$) and a resizing operation, denoted as *intp* (interpolation):



(a) Addition operation



(b) USIM with pooling

Figure 8: The Schematic diagrams for (a) resizing after the addition operation and (b) USIM

[Proof] As illustrated in Supplementary Fig. 8, the pixel values of images generated by USIM (x_{usim}) and addition operations (x_{add}) are as follows:

$$x_{usim} \in \{p_{i+n,j+m}, q_{i+n,j+m} with m, n \in \{0,1\}\} \tag{11}$$

$$\begin{aligned} x_{add} = {} & ap_{i,j} + bp_{i,j+1} + cp_{i+1,j} + dp_{i+1,j+1} \\ & + aq_{i,j} + bq_{i,j+1} + cq_{i+1,j} + dq_{i+1,j+1} \\ & (a+b+c+d=1) \end{aligned} \tag{12}$$

Since the linear interpolation is applied to the merged image generated by the addition operation, the pixel values of the newly generated pixels in intervals are considered as the results of the internal division. Eq. 12 can be calculated as a linear combination of the pixel values multiplied by the parameter values of which the sum is 1. Suppose that $p_{i,j}$ and $q_{i,j}$ are normalized as $[0,1]$, such that the bounds of x_{usim} and x_{add} are as follows:

$$x_{usim} \in [0,1] \tag{13}$$

$$x_{add} \in [0, 2(a+b+c+d)] = [0,2] \tag{14}$$

where the supremum and the infimum of $p_{i,j}$ and $q_{i,j}$ are 0 and 1, respectively. Since x_{add} is distributed in the larger range than that of x_{usim}, the pixel-level entropy of the addition operation is greater than the entropy of USIM because of Lemma I, which is described in the end of this section. However, the expectation values by USIM ($E[usim]$) can be one of $E[p_{i,j}]$, $E[p_{i+1,j}]$, $E[p_{i,j+1}]$, $E[p_{i+1,j+1}]$, $E[q_{i,j}]$, $E[q_{i+1,j}]$, $E[q_{i,j+1}]$, and $E[q_{i+1,j+1}]$, and the expectation values by addition operator ($E[add]$) can be as follows:

$$\begin{aligned} E[add] = {} & E[p_{i,j}] + E[p_{i+1,j}] + E[p_{i,j+1}] + E[p_{i+1,j+1}] \\ & + E[q_{i,j}] + E[q_{i+1,j}] + E[q_{i,j+1}] + and E[q_{i+1,j+1}] \end{aligned} \tag{15}$$

Lemma 3. The pixel-level entropy of a resized image after being merged by USIM is compared to the pixel-level entropy of the merged image by others.

[Proof] As illustrated in Figure 9, the convolution operation of which weights $w_{i,j}$ are applied to the pixel values of images generated by each operation. The generated pixel values generated by USIM (x_{usim}), the addition operation (x_{add}), and the concatenation operation (x_{concat}) are in the following:

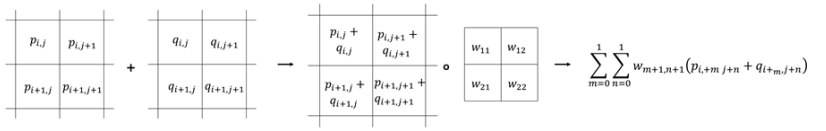
As $p_{i,j}$ and $q_{i,j}$ are normalized in $[0,1]$, the bounds of x_{usim}, x_{add}, and x_{concat} are derived when $p_{i,j} = q_{i,j} = 1$ as follows:

$$\begin{aligned} x_{usim} = {} & (w_{1,1} + w_{2,2}) f^{(i)}_{h,w,c} + (w_{2,1} + w_{1,2}) f^{(j)}_{h,w,c}) + w_{1,1} f^{(j)}_{h,w,c}) \\ & + w_{1,2} p_{i,j+1} + w_{2,1} f^{(i)}_{h,w,c} + w_{2,2} q_{i,j+1} + w_{1,2} f^{(i)}_{h,w,c} \\ & + w_{2,1} p_{i+1,j} + w_{2,2} q_{i+1,j} + w_{1,2} q_{i,j+1} + w_{2,1} q_{i+1,j} \\ & + w_{2,2} p_{i+1,j+1} \end{aligned} \tag{16}$$

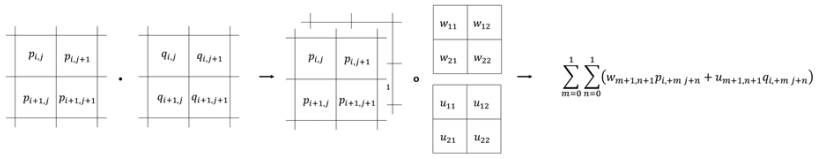
$$x_{add} = \sum_{m=0}^{1} \sum_{n=0}^{1} w_{m+1,n+1} (p_{i+m,j+n} + q_{i+m,j+n}) \tag{17}$$

$$x_{concat} = \sum_{m=0}^{1} \sum_{n=0}^{1} (w_{m+1,n+1} p_{i+m,j+n} + u_{m+1,n+1} q_{i+m,j+n}) \tag{18}$$

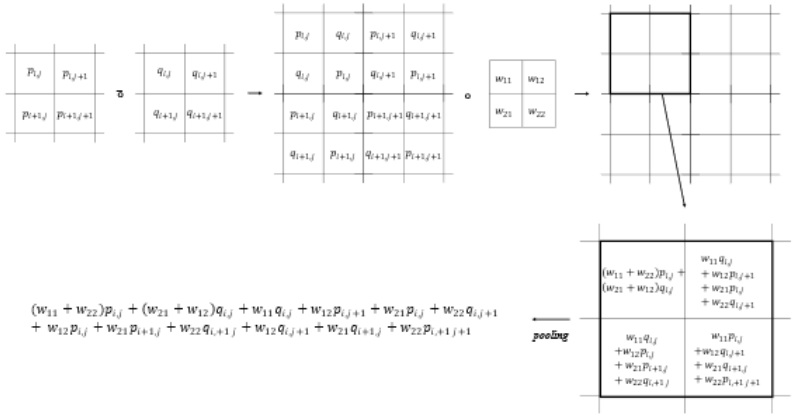
$$\begin{aligned} Supx_{usim} &= 4 \sum_{i,j} w_{i,j} \\ Supx_{add} &= 2 \sum_{i,j} w_{i,j} \\ Supx_{concat} &= 2 \sum_{i,j} w_{i,j} \end{aligned} \tag{19}$$



(a) Addition operation



(b) Concatenation operation



(c) USIM operation

Figure 9: The Schematic diagrams of (a) the addition operation with resizing operation, (b) concatenation operation, and (c) USIM.

Therefore, when using USIM, the generated pixel value will be in twice more wide range than that of other operations such that the entropy of USIM is greater than those of other operations such as the addition and concatenation operation by **Lemma 5.**, which is described in the end of this section.

Lemma 4. While the merged image by USIM is without resizing, the merged image by the addition operation or the concatenation operation is resized as the same size as the merged image by USIM. The entropy of the generated images by the convolution operation after being merged by each operation are compared.

[Proof] Since the calculation of the entropy of the generated images by the convolution operation after being merged by the operations such as USIM, the addition operation, and the concatenation operation is beyond the topics in this paper, only the experiments are conducted under this condition. In the experiment, the Shannon entropy of the generated image by the convolution operation after being merged by addition operation ($H(conv(intp(F^{(i)} + F^{(j)})))$), the Shannon entropy of the generated image by the convolution operation after being merged by the concatenation operation ($H(conv(intp(concat(F^{(i)}, F^{(j)}))))$), and the Shannon entropy of the generated image by the convolution operation after being merged by USIM ($H(conv(intp(USIM(F^{(i)}, F^{(j)}))))$) are compared through the experiment.

Lemma 5. The entropy of random variables increases as increasing the wide range of the domain of the distribution.

In this paper, USIM operation is generally utilized with convolution operations and batch normalization

[Proposition 1] The entropy of a random variable (X) which is a truncated normal distribution of $[0, a]$ is lower than the entropy of a random variable (X') which is a wide range of the truncated normal distribution of $[0, b]$, $(a < b)$.

[Prove] By definition, the probability density function of truncated normal distribution with a domain ($[a,b]$), mean (μ), and standard deviation (σ) is as follows:

$$p(x;\mu,\sigma,a,b) = \frac{1}{\sigma}\frac{\phi(\frac{x-\mu}{\sigma})}{\phi(\frac{b-\mu}{\sigma}) - \phi(\frac{a-\mu}{\sigma})} \tag{20}$$

where $\phi(x)$ is a gaussian normal distribution function and $\Phi'(x) = \phi(x)$ s.t. $\Phi(x) = \frac{1}{2}\left(1 + \sqrt{\frac{2}{\pi}}\int_0^z e^{-(\frac{x-\mu}{\sigma})^2}\right)dx$. Therefore, the entropy of the truncated normal distribution is calculated as follows:

$$\begin{aligned} H(x) &= -\int_0^a p(x)\log p(x)dx \\ &= \log\left(\sigma z\sqrt{2\pi e}\right) + \frac{\alpha\phi(\alpha) - \beta\phi(\beta)}{2z} \end{aligned} \tag{21}$$

where $\alpha = -\frac{\mu}{\sigma}$, $\beta = \frac{a-\mu}{\sigma}$, and $z = \Phi(\beta) - \Phi(\alpha)$. The subtraction of the entropy of two random variable (X) which is a truncated normal distribution of $[0,a]$ and the random variable (X') which is a wide range of the truncated normal distribution of $[0,b]$ are then as follows:

$$\Delta H = \log\frac{z_2}{z_1} + \frac{\alpha_2\phi(\alpha_2) - \beta_2\phi(\beta_2)}{2z_2} - \frac{\alpha_1\phi(\alpha_1) - \beta_1\phi(\beta_1)}{2z_1} \tag{22}$$

where $\alpha_1 = -\frac{\mu_1}{\sigma_1}$, $\beta_1 = \frac{a_1-\mu_1}{\sigma_1}$, $z_1 = \Phi(\beta_1) - \Phi(\alpha_1)$, $\alpha_2 = -\frac{\mu_2}{\sigma_2}$, $\beta_2 = \frac{b-\mu_2}{\sigma_2}$, and $z_2 = \Phi(\beta_2) - \Phi(\alpha_2)$. Here, suppose $\sigma_1 = \sigma_2$ and $\mu_1 = \mu_2$, and then $\alpha_1 = \alpha_2$. Since it is known that the function of

$$f(x) = \log\left(\Phi(\frac{x-\mu}{\sigma}) - \Phi(\alpha)\right) + \frac{\alpha\phi(\alpha) - (\frac{x-\mu}{\sigma})\phi(\frac{x-\mu}{\sigma})}{2\left(\Phi(\frac{x-\mu}{\sigma}) - \Phi(\alpha)\right)} \tag{23}$$

is a monotonic increasing function. Therefore, $\Delta H > 0$ ($\because a < b$).

As a result, by **Proposition 1**, the entropy of random variables increases as the range of the domain of the distribution widens since the entropy of the random variables, which is in the wide range of domain, is higher.

B Appendix B. Experiments

B.1 Appendix B-1. Environment Description

We utilized a server for training SoN with public datasets. The server included two CPUs of E5-2640v, 128GB RAMs, and eight Titan-Xp GPUs. Also, for precise and fast indoor positioning, we developed a deep learning-based indoor positioning system with a smartphone. The server was utilized for the training of deep learning models. After training of the SoN, an Application with the optimized SoN was implemented into the Android smartphone with a deep learning framework, TensorFlow. For the training, the batch size of the training was set to 6, and the ADAM optimizer was utilized with the default values of all parameters.

B.2 Appendix B-2. Dataset Description

To evaluate USIM, UG, and other deep learning models, we utilized four aerial image datasets of Inria, WHU, Korean Urban Dataset (KUD), and LoveDA. Here, Inria and WHU are binary building datasets, and they are utilized to validate the novel performance of USIM and UG than other operators. Meanwhile, KUD and LoveDA are multi objects segmentation datasets in aerial images, and they are utilized for the benchmark of the deep learning model with UG and other state-of-the-art models.

Inria dataset As the public data, an Inria dataset was utilized in the experiments. The Inria dataset (Dataset2) covered the area of 405 km^2. The aerial image had a spatial resolution of $0.3m$ and covers five cities (Austin, Chicago, Kitsap, Tyrol, Vienna). Here the image sets were randomly cropped into the size of 224×224 from the original size of 5000×5000, and 144,000 images were utilized for each city. Here, these images were divided into a training, a validation, and a test set for the k-fold cross validation ($k = 10$); three cities are for a training, another city is for a validation and the other city is for a test.

WHU dataset In addition, from the WHU Building Dataset, we used one of Satellite Dataset I. The dataset was collected from cities over the world and from various remote sensing resources including QuickBird, Worldview series, IKONOS, ZY-3, etc. The WHU dataset contains 204 images of which size is 224 × 224 but randomly cropped so that a total number of 20,400 images are constructed for the WHU dataset. Here, the resolutions of the WHU dataset varies from 0.3 m to 2.5 m. Likewise, these images were divided into a training, a validation, and a test set for the k-fold cross validation ($k = 10$); three cities are for a training, another city is for a validation and the other city is for a test.

Korean Urban dataset The Korean Urban Dataset (KUD) is consist of aerial images over the area of Seoul, Suwon, Anyang, Gwacheon, and Goyang. The labeled data were obtained by changing vector data provided by the government agency of National Geographic Information Institute to images using Quantum GIS, a free and open-source GIS application. These data are more accurate than the ones from the OSM because they have been made by experts for many years. As changing a data format of the labeled data, the data are labeled into four classes: background, building, road, and water. Our data set, as shown in Fig. 2, consists of pairs of RGB images with 0.51m spatial resolution and labeled images. The data set covers an area of 551km^2 and is randomly divided into an area of 486.5km^2 for training and 64.5km^2 for testing. All the data were divided into multiple images with the pixel size of 224 × 224, of which 72,400 images were assigned to the training set and 9,600 to the test set. Likewise, these images were divided into a training, a validation, and a test set for the k-fold cross-validation ($k = 10$), regarding the cities.

LoveDA dataset The LoveDA dataset covers 5,987 high spatial resolution (0.3m) remote sensing images from Nanjing, Changzhou, and Wuhan Focus on different geographical environments between Urban and Rural Advance both semantic segmentation task. The LoveDA dataset contains 2,522 images of which size is 224 × 224 but randomly cropped so that a total number of 20,000 images are constructed for the WHU dataset.

Cityscapes dataset The Cityscapes dataset focuses on semantic understanding of urban street scenes. For segmentation task, the Cityscapes dataset is consist of 30 classes (e.g., flat, human, vehicle, construction, object, nature, sky and void) from 50 cities. The dataset contains 5,000 images of which size is 1024 × 2048 and is divided into training (2,975), validation (500) and test (1,525) sets.

B.3 Appendix B-3. Experimental Results

Appendix B-3-1. Quantitative analysis

Table 5: Quantitative comparison using **Inria** dataset.

Inria	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	73.6%	88.5%	80.4%	67.2%	73.4%	72.9%	54.4%
CCB	76.6%	89.7%	82.6%	70.4%	76.2%	75.9%	60.3%
CCNet	76.7%	89.8%	82.8%	70.6%	76.4%	76.0%	60.7%
CCNet+UG	76.5%	90.5%	82.9%	70.8%	76.7%	75.9%	61.3%
CCNet+USIM	76.3%	90.3%	82.7%	70.5%	76.4%	75.6%	60.8%
DANet	73.9%	88.3%	80.5%	67.3%	73.5%	73.1%	54.5%
DANet+UG	75.7%	89.8%	82.1%	69.7%	75.7%	74.9%	59.1%
DANet+USIM	74.9%	89.2%	81.4%	68.7%	74.8%	74.1%	57.2%
DeepLabV3+	73.9%	88.8%	80.7%	67.6%	73.8%	73.2%	55.2%
DeepLabV3++UG	74.5%	89.3%	81.2%	68.4%	74.5%	73.8%	56.8%
DeepLabV3++USIM	72.4%	88.0%	79.4%	65.9%	72.3%	71.5%	52.1%
DLR9	74.9%	88.8%	81.2%	68.4%	74.5%	74.1%	56.6%
DLR9+UG	81.8%	92.7%	86.9%	76.8%	81.8%	81.1%	71.9%
DLR9+USIM	75.9%	89.5%	82.1%	69.7%	75.6%	75.2%	59.0%
ESPCN+TreeUNet	79.7%	91.7%	85.3%	74.4%	79.7%	79.1%	67.6%
FCN	69.4%	87.5%	77.4%	63.2%	70.1%	68.7%	47.5%
FCN+UG	73.9%	88.2%	80.4%	67.2%	73.4%	73.1%	54.4%
FCN+USIM	72.1%	87.9%	79.2%	65.6%	72.0%	71.5%	51.5%
FusionNet	71.3%	88.3%	78.9%	65.2%	71.8%	70.6%	51.1%
FusionNet+UG	74.5%	89.3%	81.2%	68.4%	74.5%	73.6%	56.8%

FusionNet+USIM	73.8%	88.7%	80.6%	67.4%	73.7%	73.0%	55.0%
Red-Net	77.0%	90.6%	83.2%	71.2%	77.0%	76.2%	62.0%
Red-Net+UG	83.2%	93.2%	87.9%	78.4%	83.1%	82.5%	74.7%
Red-Net+USIM	77.6%	91.1%	83.8%	72.2%	77.9%	77.0%	63.8%
RPCNet	76.4%	90.1%	82.7%	70.5%	76.4%	75.7%	60.7%
RPCNet+UG	83.3%	93.5%	88.1%	78.8%	83.4%	82.8%	75.5%
RPCNet+USIM	76.7%	90.3%	82.9%	70.8%	76.7%	76.0%	61.2%
TreeUNet	77.8%	91.0%	83.9%	72.3%	77.9%	77.1%	63.9%
TreeUNet+UG	83.4%	93.5%	88.1%	78.8%	83.4%	82.8%	75.4%
TreeUNet+USIM	79.5%	91.6%	85.1%	74.1%	79.5%	78.8%	67.2%
U-Net	72.6%	87.9%	79.5%	66.0%	72.4%	71.8%	52.3%
U-Net+UG	75.4%	89.5%	81.8%	69.3%	75.3%	74.6%	58.3%
U-Net+USIM	73.4%	88.5%	80.2%	67.0%	73.3%	72.6%	54.1%
U-NetPPL	74.7%	89.4%	81.4%	68.6%	74.7%	73.9%	57.2%
U-NetPPL+UG	77.1%	90.7%	83.3%	71.4%	77.2%	76.3%	62.4%
U-NetPPL+USIM	75.4%	89.9%	82.0%	69.5%	75.6%	74.7%	58.9%

Table 6: Quantitative comparison using **WHU** dataset.

WHU	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	67.8%	86.0%	75.9%	61.1%	69.1%	67.0%	45.4%
CCB	73.2%	86.8%	79.4%	65.9%	72.9%	71.9%	53.3%
CCNet	72.3%	87.6%	79.2%	65.6%	72.9%	71.2%	53.3%
CCNet+UG	75.0%	87.6%	80.8%	67.8%	74.5%	73.9%	56.8%
CCNet+USIM	73.0%	86.8%	79.3%	65.7%	72.8%	72.5%	53.0%
DANet	71.3%	86.0%	77.9%	63.9%	71.1%	70.4%	49.7%
DANet+UG	73.4%	87.4%	79.8%	66.3%	73.4%	72.5%	54.3%
DANet+USIM	71.5%	86.4%	78.3%	64.3%	71.6%	70.8%	50.6%
DeepLabV3+	70.4%	86.2%	77.5%	63.3%	70.8%	69.7%	48.9%
DeepLabV3++UG	71.1%	86.3%	78.0%	63.9%	71.3%	70.6%	49.9%
DeepLabV3++USIM	71.2%	86.1%	77.9%	63.8%	71.1%	70.0%	49.6%
DLR9	71.7%	87.1%	78.6%	64.8%	72.1%	71.0%	51.8%
DLR9+UG	78.7%	89.7%	83.9%	72.2%	78.3%	77.8%	64.8%
DLR9+USIM	73.2%	87.1%	79.5%	66.0%	73.0%	72.5%	53.7%
ESPCN+TreeUNet	77.4%	88.4%	82.5%	70.3%	76.6%	76.2%	61.1%
FCN	67.3%	83.7%	74.6%	59.5%	67.3%	66.4%	41.5%
FCN+UG	71.0%	86.1%	77.9%	63.7%	71.1%	70.2%	49.6%
FCN+USIM	70.6%	85.1%	77.2%	62.9%	70.2%	69.9%	47.6%
FusionNet	69.8%	85.2%	76.8%	62.3%	69.8%	68.8%	46.7%
FusionNet+UG	71.5%	87.1%	78.6%	64.7%	72.1%	70.3%	51.6%
FusionNet+USIM	70.8%	86.2%	77.7%	63.6%	71.0%	69.6%	49.3%
Red-Net	73.5%	87.0%	79.7%	66.3%	73.2%	72.5%	54.0%
Red-Net+UG	79.6%	90.4%	84.6%	73.3%	79.3%	78.8%	66.8%
Red-Net+USIM	75.9%	88.8%	81.9%	69.3%	75.9%	75.3%	59.7%
RPCNet	73.3%	86.9%	79.5%	66.0%	73.0%	72.5%	53.6%
RPCNet+UG	79.8%	89.8%	84.5%	73.2%	79.1%	79.0%	66.3%
RPCNet+USIM	74.1%	87.4%	80.2%	67.0%	73.8%	73.3%	55.3%
TreeUNet	76.2%	87.9%	81.6%	68.9%	75.4%	75.4%	58.6%
TreeUNet+UG	79.4%	89.6%	84.2%	72.7%	78.7%	78.5%	65.5%
TreeUNet+USIM	76.0%	87.9%	81.5%	68.8%	75.3%	75.2%	58.5%
U-Net	70.1%	84.8%	76.8%	62.3%	69.7%	69.2%	46.5%
U-Net+UG	71.9%	86.6%	78.6%	64.8%	72.0%	70.9%	51.4%
U-Net+USIM	68.9%	85.1%	76.2%	61.5%	69.2%	67.8%	45.5%
U-NetPPL	73.3%	87.1%	79.6%	66.1%	73.1%	72.2%	53.8%

U-NetPPL+UG	74.1%	87.0%	80.1%	66.8%	73.6%	73.4%	54.8%
U-NetPPL+USIM	72.9%	87.2%	79.4%	65.9%	73.0%	71.9%	53.5%

Table 7: Quantitative comparison using **KUD** dataset.

Background Class (0)	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	93.6%	78.5%	85.4%	74.5%	83.1%	89.6%	39.9%
CCB	94.8%	80.2%	86.9%	76.8%	84.8%	91.0%	44.7%
CCNet	93.9%	80.6%	86.7%	76.5%	84.5%	90.2%	43.3%
CCNet+UG	95.4%	83.9%	89.3%	80.6%	87.4%	92.2%	51.6%
CCNet+USIM	94.6%	81.4%	87.6%	77.9%	85.5%	91.1%	46.2%
DANet	94.0%	79.9%	86.4%	76.0%	84.2%	90.2%	42.6%
DANet+UG	94.7%	83.2%	88.6%	79.5%	86.5%	91.5%	49.0%
DANet+USIM	94.0%	81.1%	87.1%	77.1%	84.9%	90.4%	44.3%
DeepLabV3+	94.1%	78.6%	85.7%	74.9%	83.5%	90.0%	41.0%
DeepLabV3++UG	94.5%	82.5%	88.1%	78.7%	86.0%	91.2%	47.4%
DeepLabV3++USIM	93.9%	79.4%	86.1%	75.5%	83.9%	90.0%	41.8%
DLR9	94.3%	80.3%	86.7%	76.5%	84.6%	90.5%	43.7%
DLR9+UG	95.2%	83.0%	88.7%	79.6%	86.7%	91.9%	49.7%
DLR9+USIM	94.7%	81.3%	87.5%	77.8%	85.4%	91.1%	46.1%
ESPCN+TreeUNet	94.1%	82.2%	87.8%	78.2%	85.6%	90.8%	46.2%
FCN	93.3%	79.8%	86.0%	75.5%	83.8%	89.6%	41.2%
FCN+UG	94.2%	82.4%	87.9%	78.4%	85.8%	90.9%	46.7%
FCN+USIM	93.2%	79.1%	85.6%	74.8%	83.3%	89.3%	39.9%
FusionNet	94.1%	79.7%	86.3%	75.9%	84.1%	90.2%	42.5%
FusionNet+UG	94.8%	81.7%	87.8%	78.3%	85.8%	91.3%	47.0%
FusionNet+USIM	94.3%	80.9%	87.1%	77.1%	84.9%	90.7%	44.7%
Red-Net	94.7%	81.5%	87.6%	77.9%	85.5%	91.1%	46.3%
Red-Net+UG	95.3%	83.6%	89.0%	80.2%	87.1%	92.1%	50.8%
Red-Net+USIM	94.3%	82.4%	87.9%	78.5%	85.8%	91.0%	46.9%
RPCNet	94.7%	80.9%	87.3%	77.4%	85.2%	91.1%	45.5%
RPCNet+UG	95.4%	84.7%	89.7%	81.4%	87.8%	92.4%	52.9%
RPCNet+USIM	93.9%	80.1%	86.4%	76.1%	84.2%	90.1%	42.6%
TreeUNet	94.5%	80.7%	87.1%	77.1%	85.0%	90.8%	44.9%
TreeUNet+UG	95.0%	84.4%	89.4%	80.8%	87.4%	92.0%	51.4%
TreeUNet+USIM	94.6%	82.1%	87.9%	78.4%	85.8%	91.2%	47.1%
U-Net	93.5%	80.2%	86.4%	76.0%	84.1%	89.8%	42.1%
U-Net+UG	94.8%	83.0%	88.5%	79.4%	86.5%	91.5%	48.8%
U-Net+USIM	93.9%	79.6%	86.2%	75.7%	84.0%	90.1%	42.1%
U-NetPPL	94.2%	79.7%	86.4%	76.0%	84.2%	90.4%	42.9%
U-NetPPL+UG	94.6%	83.0%	88.4%	79.2%	86.4%	91.3%	48.4%
U-NetPPL+USIM	94.3%	79.4%	86.2%	75.8%	84.1%	90.4%	42.6%
Building Class (1)	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	63.3%	91.1%	74.7%	59.6%	88.2%	60.7%	32.1%
CCB	65.9%	92.5%	76.9%	62.5%	89.4%	63.5%	36.0%
CCNet	64.6%	91.7%	75.8%	61.0%	88.8%	62.1%	34.0%
CCNet+UG	69.6%	93.3%	79.7%	66.3%	90.9%	67.2%	41.4%
CCNet+USIM	66.3%	92.3%	77.2%	62.8%	89.6%	63.9%	36.5%
DANet	64.1%	91.6%	75.4%	60.6%	88.6%	61.8%	33.4%
DANet+UG	69.1%	92.1%	79.0%	65.2%	90.6%	66.5%	40.1%
DANet+USIM	66.1%	91.7%	76.8%	62.4%	89.4%	63.5%	35.9%
DeepLabV3+	63.0%	91.5%	74.6%	59.5%	88.1%	60.5%	31.8%
DeepLabV3++UG	67.6%	93.1%	78.4%	64.4%	90.2%	65.4%	38.7%

DeepLabV3++USIM	63.9%	91.2%	75.2%	60.2%	88.5%	61.3%	32.9%
DLR9	65.1%	92.1%	76.3%	61.7%	89.0%	62.7%	34.9%
DLR9+UG	68.9%	93.5%	79.3%	65.8%	90.7%	66.8%	40.7%
DLR9+USIM	65.7%	92.4%	76.8%	62.3%	89.3%	63.3%	35.7%
ESPCN+TreeUNet	66.4%	91.9%	77.1%	62.7%	89.6%	63.9%	36.5%
FCN	64.7%	90.1%	75.3%	60.4%	88.7%	61.6%	33.3%
FCN+UG	68.4%	92.0%	78.4%	64.5%	90.3%	65.7%	39.1%
FCN+USIM	63.4%	91.7%	75.0%	60.0%	88.3%	60.9%	32.5%
FusionNet	64.7%	92.1%	76.0%	61.3%	88.9%	62.4%	34.4%
FusionNet+UG	66.1%	92.6%	77.2%	62.8%	89.5%	63.9%	36.5%
FusionNet+USIM	65.9%	92.0%	76.8%	62.3%	89.4%	63.3%	35.8%
Red-Net	66.8%	92.3%	77.5%	63.3%	89.8%	64.4%	37.2%
Red-Net+UG	70.1%	93.1%	80.0%	66.6%	91.1%	67.7%	42.1%
Red-Net+USIM	68.3%	91.8%	78.3%	64.4%	90.3%	65.6%	38.9%
RPCNet	65.5%	92.1%	76.6%	62.0%	89.2%	63.0%	35.4%
RPCNet+UG	69.6%	93.0%	79.6%	66.1%	90.9%	67.2%	41.4%
RPCNet+USIM	64.4%	92.3%	75.9%	61.1%	88.8%	62.2%	34.1%
TreeUNet	65.7%	92.1%	76.7%	62.2%	89.3%	63.4%	35.7%
TreeUNet+UG	69.9%	92.9%	79.8%	66.4%	91.0%	67.4%	41.7%
TreeUNet+USIM	66.2%	92.7%	77.3%	62.9%	89.6%	64.0%	36.7%
U-Net	64.5%	91.0%	75.5%	60.6%	88.7%	61.8%	33.5%
U-Net+UG	68.4%	92.2%	78.5%	64.6%	90.3%	65.8%	39.2%
U-Net+USIM	63.0%	90.9%	74.4%	59.3%	88.0%	60.4%	31.7%
U-NetPPL	64.6%	92.0%	75.9%	61.2%	88.8%	62.2%	34.2%
U-NetPPL+UG	68.1%	92.5%	78.5%	64.6%	90.3%	65.6%	39.0%
U-NetPPL+USIM	63.2%	92.2%	75.0%	60.0%	88.2%	60.8%	32.4%
Road Class (2)	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	68.0%	71.3%	69.6%	53.4%	90.4%	56.4%	29.2%
CCB	70.0%	74.4%	72.1%	56.4%	91.2%	59.3%	32.7%
CCNet	71.0%	70.6%	70.8%	54.8%	91.1%	58.4%	31.4%
CCNet+UG	75.3%	76.2%	75.7%	60.9%	92.5%	64.3%	39.0%
CCNet+USIM	71.8%	73.0%	72.4%	56.7%	91.5%	60.0%	33.5%
DANet	70.2%	71.7%	70.9%	54.9%	91.0%	58.3%	31.3%
DANet+UG	73.9%	75.9%	74.9%	59.8%	92.2%	63.1%	37.4%
DANet+USIM	69.9%	71.0%	70.4%	54.4%	90.8%	57.8%	30.7%
DeepLabV3+	69.6%	72.5%	71.0%	55.1%	90.9%	58.2%	31.3%
DeepLabV3++UG	73.9%	73.6%	73.8%	58.4%	92.0%	62.0%	35.9%
DeepLabV3++USIM	69.2%	71.4%	70.2%	54.1%	90.7%	57.4%	30.2%
DLR9	71.6%	73.7%	72.6%	57.0%	91.5%	60.2%	33.8%
DLR9+UG	75.1%	77.4%	76.2%	61.6%	92.6%	64.8%	39.7%
DLR9+USIM	73.0%	73.8%	73.4%	58.0%	91.8%	61.3%	35.1%
ESPCN+TreeUNet	73.1%	72.0%	72.5%	56.9%	91.6%	60.6%	34.1%
FCN	68.3%	70.9%	69.6%	53.4%	90.5%	56.5%	29.3%
FCN+UG	71.9%	74.1%	72.9%	57.4%	91.6%	60.6%	34.2%
FCN+USIM	67.6%	67.2%	67.4%	50.8%	90.0%	54.3%	26.7%
FusionNet	69.9%	72.1%	71.0%	55.0%	91.0%	58.3%	31.3%
FusionNet+UG	74.3%	75.1%	74.7%	59.6%	92.2%	63.0%	37.3%
FusionNet+USIM	71.8%	73.8%	72.7%	57.2%	91.5%	60.4%	34.0%
Red-Net	71.2%	74.0%	72.6%	56.9%	91.4%	60.0%	33.6%
Red-Net+UG	75.0%	78.3%	76.6%	62.1%	92.7%	65.0%	40.1%
Red-Net+USIM	70.8%	73.1%	71.9%	56.1%	91.2%	59.3%	32.6%
RPCNet	70.6%	73.6%	72.0%	56.3%	91.2%	59.4%	32.8%
RPCNet+UG	77.4%	76.7%	77.1%	62.7%	93.0%	66.2%	41.5%
RPCNet+USIM	70.7%	70.8%	70.7%	54.7%	91.0%	58.1%	31.2%
TreeUNet	71.4%	74.0%	72.6%	57.0%	91.5%	60.1%	33.7%

TreeUNet+UG	76.6%	76.4%	76.5%	61.9%	92.8%	65.5%	40.5%
TreeUNet+USIM	73.3%	72.8%	73.0%	57.5%	91.7%	61.0%	34.7%
U-Net	70.8%	71.6%	71.2%	55.3%	91.1%	58.8%	31.9%
U-Net+UG	73.5%	74.7%	74.1%	58.9%	92.0%	62.1%	36.2%
U-Net+USIM	70.7%	70.6%	70.7%	54.7%	91.0%	58.2%	31.2%
U-NetPPL	69.7%	73.3%	71.5%	55.6%	91.0%	58.6%	31.8%
U-NetPPL+UG	74.2%	73.8%	74.0%	58.7%	92.0%	62.3%	36.3%
U-NetPPL+USIM	71.4%	72.2%	71.8%	56.0%	91.3%	59.4%	32.7%
Water Class (3)	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	65.7%	88.8%	75.5%	60.7%	98.4%	59.4%	43.5%
CCB	65.7%	88.7%	75.5%	60.6%	98.4%	59.3%	43.4%
CCNet	66.4%	88.7%	75.9%	61.2%	98.4%	60.0%	44.1%
CCNet+UG	70.0%	90.9%	79.1%	65.4%	98.7%	64.6%	49.7%
CCNet+USIM	66.5%	90.0%	76.5%	61.9%	98.5%	60.9%	45.2%
DANet	67.3%	87.6%	76.1%	61.4%	98.5%	60.1%	44.3%
DANet+UG	68.3%	90.3%	77.8%	63.6%	98.6%	62.6%	47.4%
DANet+USIM	66.3%	88.2%	75.7%	60.9%	98.4%	59.6%	43.7%
DeepLabV3+	65.9%	89.2%	75.8%	61.0%	98.4%	59.8%	44.0%
DeepLabV3++UG	68.5%	89.6%	77.7%	63.5%	98.6%	62.4%	47.1%
DeepLabV3++USIM	64.9%	87.7%	74.6%	59.5%	98.3%	58.1%	41.9%
DLR9	66.4%	89.1%	76.1%	61.4%	98.4%	60.3%	44.5%
DLR9+UG	70.8%	90.8%	79.5%	66.0%	98.7%	65.2%	50.5%
DLR9+USIM	67.1%	88.8%	76.5%	61.9%	98.5%	60.7%	45.0%
ESPCN+TreeUNet	68.6%	88.9%	77.5%	63.2%	98.6%	62.2%	46.7%
FCN	65.4%	87.8%	74.9%	59.9%	98.4%	58.5%	42.4%
FCN+UG	68.9%	88.4%	77.4%	63.2%	98.6%	62.1%	46.6%
FCN+USIM	64.3%	88.1%	74.4%	59.2%	98.3%	57.8%	41.6%
FusionNet	65.3%	88.2%	75.0%	60.0%	98.4%	58.7%	42.7%
FusionNet+UG	68.6%	89.4%	77.6%	63.4%	98.6%	62.4%	47.0%
FusionNet+USIM	66.0%	88.3%	75.6%	60.7%	98.4%	59.4%	43.5%
Red-Net	68.1%	88.4%	76.9%	62.5%	98.5%	61.3%	45.7%
Red-Net+UG	71.8%	90.1%	79.9%	66.5%	98.7%	65.6%	51.0%
Red-Net+USIM	67.5%	89.1%	76.8%	62.3%	98.5%	61.3%	45.7%
RPCNet	70.0%	88.7%	78.2%	64.3%	98.6%	63.2%	48.0%
RPCNet+UG	72.3%	90.2%	80.3%	67.0%	98.8%	66.2%	51.7%
RPCNet+USIM	66.7%	88.5%	76.1%	61.4%	98.4%	60.2%	44.4%
TreeUNet	66.4%	89.1%	76.1%	61.4%	98.4%	60.2%	44.4%
TreeUNet+UG	71.3%	91.1%	80.0%	66.7%	98.7%	65.8%	51.3%
TreeUNet+USIM	69.0%	88.9%	77.7%	63.5%	98.6%	62.4%	47.0%
U-Net	66.7%	87.7%	75.8%	61.0%	98.4%	59.6%	43.7%
U-Net+UG	67.8%	89.1%	77.0%	62.6%	98.5%	61.6%	46.0%
U-Net+USIM	63.4%	87.6%	73.6%	58.2%	98.2%	56.8%	40.4%
U-NetPPL	69.1%	88.7%	77.7%	63.5%	98.6%	62.5%	47.1%
U-NetPPL+UG	68.3%	90.0%	77.7%	63.5%	98.6%	62.4%	47.1%
U-NetPPL+USIM	66.0%	88.1%	75.5%	60.6%	98.4%	59.3%	43.3%

Table 8: Quantitative comparison using **LoveDA** dataset.

Agriculture Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	66.3%	78.0%	71.6%	55.8%	90.3%	58.0%	31.0%
CCB	68.7%	78.8%	73.4%	58.0%	91.0%	60.2%	33.7%
CCNet	68.0%	79.1%	73.1%	57.6%	90.9%	59.8%	33.2%
CCNet+UG	**74.4%**	81.2%	77.6%	63.4%	**92.7%**	**66.0%**	**41.3%**

CCNet+USIM	68.4%	78.8%	73.2%	57.8%	91.0%	60.0%	33.5%
DANet	66.4%	78.7%	72.0%	56.3%	90.4%	58.4%	31.5%
DANet+UG	70.3%	81.3%	75.4%	60.5%	91.7%	62.6%	36.9%
DANet+USIM	68.4%	78.0%	72.9%	57.3%	90.9%	59.8%	33.1%
DeepLabV3+	67.6%	78.0%	72.4%	56.8%	90.7%	59.0%	32.2%
DeepLabV3++UG	69.3%	82.0%	75.1%	60.1%	91.5%	62.1%	36.2%
DeepLabV3++USIM	68.0%	77.8%	72.6%	57.0%	90.8%	59.3%	32.5%
DLR9	67.0%	78.8%	72.4%	56.8%	90.6%	58.9%	32.1%
DLR9+UG	72.7%	*82.4%*	77.3%	62.9%	92.4%	65.1%	40.3%
DLR9+USIM	70.9%	78.3%	74.4%	59.3%	91.6%	61.9%	35.8%
ESPCN+TreeUNet	69.8%	80.8%	74.9%	59.8%	91.5%	62.0%	36.1%
FCN	66.4%	77.2%	71.4%	55.5%	90.3%	57.8%	30.6%
FCN+UG	70.2%	81.1%	75.3%	60.3%	91.6%	62.5%	36.7%
FCN+USIM	67.8%	78.6%	72.8%	57.3%	90.8%	59.4%	32.8%
FusionNet	65.8%	77.4%	71.1%	55.2%	90.2%	57.3%	30.2%
FusionNet+UG	70.5%	79.7%	74.8%	59.8%	91.6%	62.3%	36.2%
FusionNet+USIM	69.5%	78.0%	73.5%	58.1%	91.2%	60.5%	34.1%
Red-Net	68.5%	79.2%	73.5%	58.1%	91.0%	60.3%	33.8%
Red-Net+UG	*73.5%*	82.1%	77.6%	63.3%	*92.5%*	*65.7%*	*41.0%*
Red-Net+USIM	71.6%	80.7%	75.8%	61.1%	91.9%	63.4%	37.9%
RPCNet	69.2%	79.2%	73.9%	58.6%	91.2%	61.0%	34.6%
RPCNet+UG	73.2%	82.0%	77.4%	63.1%	92.5%	65.4%	40.6%
RPCNet+USIM	69.2%	80.6%	74.4%	59.3%	91.3%	61.4%	35.3%
TreeUNet	70.5%	79.6%	74.8%	59.7%	91.6%	62.0%	36.1%
TreeUNet+UG	73.1%	**82.8%**	**77.7%**	**63.5%**	92.5%	65.7%	41.0%
TreeUNet+USIM	71.1%	81.2%	75.8%	61.1%	91.9%	63.3%	37.8%
U-Net	67.8%	78.2%	72.7%	57.1%	90.8%	59.3%	32.6%
U-Net+UG	72.5%	80.9%	76.5%	61.9%	92.2%	64.3%	39.1%
U-Net+USIM	67.5%	77.1%	72.0%	56.2%	90.6%	58.6%	31.7%
U-NetPPL	70.1%	79.7%	74.6%	59.5%	91.5%	61.8%	35.7%
U-NetPPL+UG	72.8%	81.1%	76.7%	62.3%	92.3%	64.6%	39.5%
U-NetPPL+USIM	66.8%	79.2%	72.5%	56.9%	90.6%	58.9%	32.1%
Background Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	52.2%	86.9%	65.2%	48.4%	96.0%	46.9%	28.3%
CCB	54.6%	88.1%	67.5%	50.9%	96.4%	49.7%	31.2%
CCNet	53.3%	88.1%	66.4%	49.7%	96.2%	48.6%	30.0%
CCNet+UG	57.7%	89.1%	70.1%	53.9%	96.8%	52.8%	34.7%
CCNet+USIM	53.0%	87.3%	65.9%	49.2%	96.1%	47.8%	29.3%
DANet	51.5%	87.5%	64.9%	48.0%	96.0%	46.7%	28.0%
DANet+UG	57.5%	89.3%	70.0%	53.8%	96.7%	52.8%	34.6%
DANet+USIM	53.4%	87.5%	66.3%	49.6%	96.2%	48.4%	29.8%
DeepLabV3+	53.3%	87.2%	66.2%	49.4%	96.2%	48.1%	29.6%
DeepLabV3++UG	56.4%	89.3%	69.1%	52.8%	96.6%	51.7%	33.5%
DeepLabV3++USIM	51.5%	87.7%	64.9%	48.0%	95.9%	46.6%	28.0%
DLR9	53.7%	87.2%	66.5%	49.8%	96.2%	48.4%	29.9%
DLR9+UG	60.6%	89.8%	72.4%	56.7%	97.1%	55.8%	38.0%
DLR9+USIM	54.0%	88.0%	66.9%	50.3%	96.3%	49.1%	30.6%
ESPCN+TreeUNet	55.8%	88.9%	68.6%	52.2%	96.5%	51.0%	32.7%
FCN	50.4%	86.1%	63.6%	46.6%	95.8%	45.1%	26.4%
FCN+UG	55.5%	88.2%	68.1%	51.6%	96.5%	50.4%	32.0%
FCN+USIM	50.3%	87.2%	63.8%	46.8%	95.8%	45.5%	26.8%
FusionNet	51.1%	86.6%	64.3%	47.4%	95.9%	46.0%	27.3%
FusionNet+UG	55.3%	88.8%	68.2%	51.7%	96.5%	50.5%	32.1%
FusionNet+USIM	53.1%	87.6%	66.1%	49.4%	96.2%	48.2%	29.6%
Red-Net	56.4%	88.0%	68.7%	52.4%	96.6%	51.1%	32.8%

Red-Net+UG	58.5%	90.0%	70.9%	55.0%	96.9%	54.0%	36.0%
Red-Net+USIM	53.6%	88.4%	66.7%	50.1%	96.2%	48.9%	30.4%
RPCNet	53.7%	88.4%	66.8%	50.1%	96.2%	48.9%	30.4%
RPCNet+UG	59.4%	89.8%	71.5%	55.6%	96.9%	54.6%	36.7%
RPCNet+USIM	55.0%	88.1%	67.7%	51.2%	96.4%	49.9%	31.5%
TreeUNet	55.0%	88.6%	67.9%	51.4%	96.4%	50.3%	31.8%
TreeUNet+UG	60.6%	89.3%	72.2%	56.5%	97.1%	55.5%	37.7%
TreeUNet+USIM	56.2%	89.2%	68.9%	52.6%	96.6%	51.4%	33.1%
U-Net	50.8%	86.7%	64.1%	47.2%	95.9%	45.6%	27.0%
U-Net+UG	54.3%	88.4%	67.3%	50.7%	96.3%	49.4%	31.0%
U-Net+USIM	51.9%	87.2%	65.0%	48.2%	96.0%	46.8%	28.2%
U-NetPPL	54.2%	88.1%	67.2%	50.6%	96.3%	49.4%	30.9%
U-NetPPL+UG	58.3%	89.1%	70.4%	54.4%	96.8%	53.2%	35.2%
U-NetPPL+USIM	54.8%	88.3%	67.6%	51.1%	96.4%	49.9%	31.4%
Barren Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	59.2%	80.4%	68.2%	51.7%	91.9%	52.0%	28.2%
CCB	60.6%	81.2%	69.4%	53.2%	92.2%	53.4%	29.8%
CCNet	60.6%	80.4%	69.1%	52.8%	92.2%	53.1%	29.4%
CCNet+UG	61.6%	83.0%	70.7%	54.7%	92.6%	54.9%	31.5%
CCNet+USIM	60.1%	81.0%	69.0%	52.6%	92.1%	52.9%	29.2%
DANet	56.1%	80.9%	66.3%	49.6%	91.1%	49.4%	25.5%
DANet+UG	59.1%	81.9%	68.6%	52.3%	91.9%	52.4%	28.7%
DANet+USIM	57.3%	81.0%	67.1%	50.5%	91.4%	50.6%	26.7%
DeepLabV3+	59.1%	80.6%	68.2%	51.8%	91.9%	51.9%	28.1%
DeepLabV3++UG	60.5%	82.9%	69.9%	53.8%	92.3%	54.0%	30.4%
DeepLabV3++USIM	55.5%	80.1%	65.6%	48.8%	90.9%	48.7%	24.7%
DLR9	58.4%	80.8%	67.8%	51.3%	91.7%	51.5%	27.6%
DLR9+UG	63.3%	84.8%	72.5%	56.9%	93.0%	57.3%	34.3%
DLR9+USIM	58.8%	81.7%	68.4%	52.0%	91.8%	52.2%	28.3%
ESPCN+TreeUNet	61.7%	81.9%	70.4%	54.3%	92.5%	54.7%	31.2%
FCN	55.4%	79.4%	65.3%	48.5%	90.8%	48.5%	24.4%
FCN+UG	60.9%	81.6%	69.7%	53.5%	92.3%	53.9%	30.3%
FCN+USIM	57.8%	79.6%	66.9%	50.3%	91.5%	50.5%	26.5%
FusionNet	55.9%	80.2%	65.8%	49.1%	91.0%	49.1%	25.0%
FusionNet+UG	62.8%	81.9%	71.1%	55.2%	92.8%	55.5%	32.2%
FusionNet+USIM	58.8%	80.9%	68.1%	51.6%	91.8%	51.8%	28.0%
Red-Net	60.0%	81.6%	69.2%	52.9%	92.1%	53.1%	29.4%
Red-Net+UG	63.6%	84.5%	72.6%	57.0%	93.1%	57.3%	34.4%
Red-Net+USIM	59.0%	80.8%	68.2%	51.8%	91.8%	51.8%	28.0%
RPCNet	61.7%	81.2%	70.1%	54.0%	92.5%	54.4%	30.8%
RPCNet+UG	63.0%	83.2%	71.7%	55.9%	92.9%	56.2%	33.0%
RPCNet+USIM	59.8%	81.6%	69.0%	52.7%	92.1%	52.9%	29.2%
TreeUNet	62.1%	81.2%	70.4%	54.3%	92.6%	54.6%	31.2%
TreeUNet+UG	65.9%	84.1%	73.9%	58.6%	93.6%	59.0%	36.5%
TreeUNet+USIM	59.7%	82.4%	69.3%	53.0%	92.1%	53.2%	29.5%
U-Net	58.4%	80.3%	67.6%	51.1%	91.7%	51.3%	27.3%
U-Net+UG	61.3%	82.5%	70.3%	54.2%	92.4%	54.4%	31.0%
U-Net+USIM	57.6%	79.9%	66.9%	50.3%	91.4%	50.5%	26.5%
U-NetPPL	58.0%	81.1%	67.6%	51.1%	91.6%	51.3%	27.4%
U-NetPPL+UG	61.0%	83.4%	70.5%	54.4%	92.4%	54.7%	31.2%
U-NetPPL+USIM	58.8%	80.4%	67.9%	51.4%	91.8%	51.5%	27.6%
Building Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	88.8%	35.8%	51.1%	34.3%	69.6%	67.0%	11.0%
CCB	91.0%	40.9%	56.4%	39.3%	72.0%	70.3%	14.5%
CCNet	90.8%	41.5%	57.0%	39.8%	72.2%	70.4%	14.8%

CCNet+UG	92.3%	46.8%	62.1%	45.0%	74.7%	73.6%	18.9%
CCNet+USIM	90.0%	38.3%	53.7%	36.7%	70.8%	68.7%	12.7%
DANet	87.7%	32.5%	47.4%	31.0%	68.1%	64.9%	9.1%
DANet+UG	91.5%	41.5%	57.1%	39.9%	72.4%	70.8%	15.1%
DANet+USIM	89.8%	37.3%	52.7%	35.8%	70.4%	68.1%	12.1%
DeepLabV3+	88.9%	35.9%	51.1%	34.3%	69.6%	67.1%	11.0%
DeepLabV3++UG	91.4%	41.7%	57.2%	40.1%	72.4%	70.8%	15.2%
DeepLabV3++USIM	88.5%	33.1%	48.2%	31.7%	68.5%	65.7%	9.6%
DLR9	88.8%	36.3%	51.5%	34.7%	69.8%	67.2%	11.2%
DLR9+UG	93.2%	47.7%	63.1%	46.1%	75.3%	74.5%	20.1%
DLR9+USIM	90.2%	37.1%	52.6%	35.7%	70.4%	68.2%	12.1%
ESPCN+TreeUNet	91.4%	42.0%	57.6%	40.4%	72.6%	71.0%	15.4%
FCN	87.3%	31.1%	45.8%	29.7%	67.5%	64.2%	8.4%
FCN+UG	91.4%	41.9%	57.5%	40.3%	72.5%	70.9%	15.3%
FCN+USIM	88.2%	33.0%	48.1%	31.6%	68.4%	65.4%	9.5%
FusionNet	88.3%	33.3%	48.3%	31.9%	68.5%	65.6%	9.6%
FusionNet+UG	91.8%	44.6%	60.0%	42.9%	73.7%	72.3%	17.2%
FusionNet+USIM	89.5%	37.0%	52.3%	35.4%	70.2%	67.8%	11.8%
Red-Net	90.9%	39.4%	55.0%	37.9%	71.4%	69.6%	13.6%
Red-Net+UG	93.3%	49.7%	64.8%	47.9%	76.1%	75.3%	21.6%
Red-Net+USIM	90.8%	40.2%	55.8%	38.7%	71.7%	69.9%	14.1%
RPCNet	91.0%	42.1%	57.5%	40.4%	72.5%	70.7%	15.2%
RPCNet+UG	93.1%	47.8%	63.2%	46.2%	75.3%	74.4%	20.1%
RPCNet+USIM	90.0%	36.6%	52.1%	35.2%	70.1%	67.9%	11.8%
TreeUNet	91.7%	45.1%	60.5%	43.4%	73.9%	72.5%	17.4%
TreeUNet+UG	93.3%	49.7%	64.9%	48.0%	76.2%	75.4%	21.6%
TreeUNet+USIM	90.9%	39.9%	55.5%	38.4%	71.6%	69.8%	13.9%
U-Net	88.9%	37.2%	52.4%	35.5%	70.1%	67.6%	11.7%
U-Net+UG	91.5%	43.2%	58.7%	41.5%	73.1%	71.6%	16.1%
U-Net+USIM	89.0%	36.3%	51.6%	34.7%	69.8%	67.3%	11.2%
U-NetPPL	90.4%	38.1%	53.6%	36.6%	70.8%	68.7%	12.7%
U-NetPPL+UG	92.2%	44.6%	60.2%	43.0%	73.8%	72.5%	17.4%
U-NetPPL+USIM	90.4%	38.4%	53.9%	36.9%	70.9%	68.9%	12.9%

Forest Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	42.8%	85.0%	56.9%	39.8%	92.1%	38.5%	18.4%
CCB	44.9%	85.8%	59.0%	41.8%	92.6%	40.8%	20.5%
CCNet	48.2%	85.7%	61.7%	44.6%	93.4%	43.3%	23.1%
CCNet+UG	50.2%	87.3%	63.7%	46.8%	93.9%	45.8%	25.5%
CCNet+USIM	46.7%	85.6%	60.5%	43.3%	93.1%	42.2%	21.9%
DANet	45.1%	84.8%	58.9%	41.7%	92.7%	40.5%	20.2%
DANet+UG	49.0%	87.5%	62.8%	45.8%	93.6%	44.8%	24.5%
DANet+USIM	45.2%	85.4%	59.1%	41.9%	92.7%	40.7%	20.4%
DeepLabV3+	43.8%	86.1%	58.1%	40.9%	92.3%	39.8%	19.6%
DeepLabV3++UG	50.0%	87.5%	63.6%	46.7%	93.8%	45.7%	25.4%
DeepLabV3++USIM	43.9%	86.5%	58.3%	41.1%	92.3%	40.0%	19.8%
DLR9	46.2%	85.6%	60.0%	42.9%	93.0%	41.9%	21.5%
DLR9+UG	52.7%	88.6%	66.0%	49.3%	94.4%	48.5%	28.3%
DLR9+USIM	44.0%	85.8%	58.1%	41.0%	92.4%	39.9%	19.7%
ESPCN+TreeUNet	47.9%	87.0%	61.8%	44.7%	93.3%	43.8%	23.4%
FCN	43.3%	85.1%	57.4%	40.2%	92.2%	39.2%	18.9%
FCN+UG	48.9%	86.9%	62.6%	45.5%	93.6%	44.6%	24.2%
FCN+USIM	44.3%	85.1%	58.3%	41.1%	92.5%	40.0%	19.7%
FusionNet	45.6%	84.2%	59.2%	42.1%	92.8%	40.8%	20.6%
FusionNet+UG	45.5%	86.3%	59.6%	42.5%	92.8%	41.4%	21.1%
FusionNet+USIM	43.9%	85.8%	58.1%	41.0%	92.4%	39.9%	19.6%

Red-Net	47.4%	86.0%	61.1%	44.0%	93.2%	43.0%	22.6%
Red-Net+UG	52.6%	88.5%	66.0%	49.3%	94.4%	48.4%	28.2%
Red-Net+USIM	48.0%	86.6%	61.8%	44.7%	93.4%	43.7%	23.3%
RPCNet	47.6%	87.0%	61.5%	44.4%	93.3%	43.5%	23.1%
RPCNet+UG	50.4%	88.1%	64.1%	47.2%	93.9%	46.4%	26.1%
RPCNet+USIM	45.9%	86.5%	59.9%	42.8%	92.9%	41.7%	21.4%
TreeUNet	48.7%	85.9%	62.1%	45.1%	93.5%	44.1%	23.7%
TreeUNet+UG	52.1%	88.5%	65.6%	48.8%	94.3%	48.1%	27.8%
TreeUNet+USIM	47.1%	87.2%	61.2%	44.1%	93.2%	43.0%	22.7%
U-Net	45.9%	85.3%	59.7%	42.6%	92.9%	41.6%	21.2%
U-Net+UG	47.7%	86.9%	61.6%	44.5%	93.3%	43.6%	23.2%
U-Net+USIM	43.6%	85.4%	57.8%	40.6%	92.3%	39.5%	19.3%
U-NetPPL	46.7%	85.8%	60.5%	43.4%	93.1%	42.3%	22.0%
U-NetPPL+UG	49.7%	87.3%	63.3%	46.3%	93.7%	45.5%	25.1%
U-NetPPL+USIM	45.7%	85.3%	59.5%	42.4%	92.8%	41.1%	20.9%
Road Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	51.9%	77.5%	62.1%	45.1%	88.3%	45.6%	20.0%
CCB	54.5%	79.5%	64.7%	47.8%	89.3%	48.2%	22.7%
CCNet	53.6%	78.7%	63.8%	46.8%	89.0%	47.3%	21.8%
CCNet+UG	54.5%	80.2%	64.9%	48.1%	89.3%	48.3%	22.9%
CCNet+USIM	51.9%	79.0%	62.6%	45.6%	88.4%	46.0%	20.4%
DANet	48.8%	77.3%	59.8%	42.7%	87.2%	42.8%	17.4%
DANet+UG	53.0%	80.8%	64.0%	47.1%	88.8%	47.5%	21.9%
DANet+USIM	52.6%	78.7%	63.1%	46.1%	88.6%	46.5%	20.9%
DeepLabV3+	50.6%	77.2%	61.2%	44.1%	87.9%	44.4%	18.9%
DeepLabV3++UG	54.3%	80.0%	64.7%	47.8%	89.2%	48.2%	22.8%
DeepLabV3++USIM	49.4%	78.1%	60.5%	43.4%	87.4%	43.8%	18.2%
DLR9	50.2%	77.0%	60.8%	43.7%	87.7%	43.8%	18.4%
DLR9+UG	56.6%	82.3%	67.1%	50.5%	90.0%	51.1%	25.7%
DLR9+USIM	50.9%	79.7%	62.1%	45.0%	88.0%	45.1%	19.7%
ESPCN+TreeUNet	53.7%	80.5%	64.4%	47.5%	89.0%	47.9%	22.4%
FCN	48.5%	77.2%	59.6%	42.4%	87.1%	42.5%	17.1%
FCN+UG	52.5%	80.0%	63.4%	46.4%	88.6%	46.8%	21.2%
FCN+USIM	48.0%	77.2%	59.2%	42.0%	86.8%	42.4%	16.9%
FusionNet	49.9%	77.6%	60.8%	43.6%	87.6%	44.0%	18.4%
FusionNet+UG	55.8%	79.1%	65.5%	48.7%	89.7%	49.1%	23.7%
FusionNet+USIM	50.9%	78.3%	61.7%	44.6%	88.0%	44.8%	19.3%
Red-Net	52.4%	80.0%	63.3%	46.3%	88.6%	46.5%	21.1%
Red-Net+UG	58.9%	82.1%	68.6%	52.1%	90.7%	52.6%	27.6%
Red-Net+USIM	52.2%	79.3%	62.9%	45.9%	88.5%	46.2%	20.7%
RPCNet	54.3%	79.3%	64.5%	47.6%	89.2%	48.1%	22.6%
RPCNet+UG	55.7%	82.4%	66.5%	49.8%	89.7%	50.0%	24.8%
RPCNet+USIM	49.9%	79.9%	61.4%	44.3%	87.6%	44.5%	18.9%
TreeUNet	53.6%	79.1%	63.9%	47.0%	89.0%	47.5%	21.9%
TreeUNet+UG	57.8%	82.1%	67.8%	51.3%	90.4%	51.8%	26.6%
TreeUNet+USIM	53.3%	80.5%	64.1%	47.2%	88.9%	47.5%	22.0%
U-Net	50.8%	77.0%	61.2%	44.1%	87.9%	44.4%	18.9%
U-Net+UG	52.2%	79.0%	62.8%	45.8%	88.5%	46.1%	20.6%
U-Net+USIM	52.7%	77.1%	62.6%	45.6%	88.6%	45.8%	20.4%
U-NetPPL	50.9%	79.1%	61.9%	44.9%	88.0%	45.0%	19.5%
U-NetPPL+UG	53.9%	80.8%	64.7%	47.8%	89.1%	48.1%	22.6%
U-NetPPL+USIM	54.1%	79.0%	64.2%	47.3%	89.1%	47.7%	22.2%
Water Class	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	48.5%	84.4%	61.6%	44.5%	93.3%	43.3%	22.8%
CCB	50.6%	86.0%	63.7%	46.7%	93.7%	45.6%	25.2%

CCNet	50.4%	86.2%	63.6%	46.6%	93.7%	45.6%	25.1%
CCNet+UG	54.7%	88.0%	67.4%	50.9%	94.6%	50.1%	29.8%
CCNet+USIM	48.6%	85.7%	62.0%	44.9%	93.3%	43.8%	23.3%
DANet	49.8%	85.3%	62.9%	45.9%	93.6%	44.7%	24.2%
DANet+UG	54.5%	87.4%	67.1%	50.5%	94.5%	49.8%	29.5%
DANet+USIM	49.4%	86.9%	63.0%	46.0%	93.5%	45.1%	24.5%
DeepLabV3+	48.4%	85.5%	61.8%	44.8%	93.3%	43.7%	23.1%
DeepLabV3++UG	52.4%	87.5%	65.6%	48.8%	94.1%	48.0%	27.6%
DeepLabV3++USIM	49.9%	85.2%	63.0%	46.0%	93.6%	44.9%	24.4%
DLR9	48.8%	84.9%	62.0%	44.9%	93.3%	43.9%	23.4%
DLR9+UG	54.4%	88.4%	67.3%	50.7%	94.5%	49.9%	29.7%
DLR9+USIM	50.0%	86.2%	63.3%	46.3%	93.6%	45.1%	24.7%
ESPCN+TreeUNet	52.3%	86.4%	65.2%	48.3%	94.1%	47.3%	26.9%
FCN	48.2%	85.3%	61.6%	44.5%	93.2%	43.3%	22.8%
FCN+UG	53.1%	86.6%	65.8%	49.0%	94.2%	48.1%	27.7%
FCN+USIM	49.0%	85.3%	62.3%	45.2%	93.4%	44.3%	23.7%
FusionNet	47.2%	84.7%	60.6%	43.5%	93.0%	42.4%	21.9%
FusionNet+UG	52.7%	86.9%	65.6%	48.9%	94.2%	48.1%	27.6%
FusionNet+USIM	49.4%	85.8%	62.7%	45.7%	93.5%	44.7%	24.2%
Red-Net	49.0%	86.1%	62.5%	45.4%	93.4%	44.2%	23.8%
Red-Net+UG	54.8%	88.3%	67.6%	51.1%	94.6%	50.3%	30.1%
Red-Net+USIM	50.8%	86.8%	64.1%	47.2%	93.8%	46.3%	25.8%
RPCNet	50.6%	86.3%	63.8%	46.8%	93.7%	45.8%	25.3%
RPCNet+UG	57.6%	88.5%	69.8%	53.6%	95.1%	52.9%	32.9%
RPCNet+USIM	52.0%	86.6%	65.0%	48.1%	94.0%	47.1%	26.7%
TreeUNet	53.2%	85.8%	65.7%	48.9%	94.3%	47.8%	27.4%
TreeUNet+UG	53.7%	87.7%	66.6%	49.9%	94.4%	49.1%	28.7%
TreeUNet+USIM	51.3%	87.0%	64.6%	47.7%	93.9%	46.8%	26.3%
U-Net	49.4%	84.9%	62.4%	45.4%	93.5%	44.4%	23.8%
U-Net+UG	55.0%	87.0%	67.4%	50.8%	94.6%	50.0%	29.7%
U-Net+USIM	47.1%	85.6%	60.7%	43.6%	92.9%	42.5%	22.0%
U-NetPPL	50.2%	85.5%	63.3%	46.3%	93.6%	45.2%	24.7%
U-NetPPL+UG	53.8%	86.7%	66.4%	49.7%	94.4%	48.7%	28.4%
U-NetPPL+USIM	48.3%	85.6%	61.8%	44.7%	93.2%	43.7%	23.2%

Table 9: Quantitative comparison using **CityScape** dataset.

Inria	precision	recall	F1 score	IoU	accuracy	AP	BJ
AU-Net	55.88%	56.23%	56.86%	48.26%	50.92%	46.37%	38.82%
CCB	52.22%	61.44%	59.65%	51.44%	57.50%	47.94%	39.39%
CCNet	57.60%	61.60%	52.97%	52.13%	51.66%	50.81%	39.55%
CCNet+UG	49.94%	57.96%	60.22%	46.49%	56.08%	55.82%	42.06%
CCNet+USIM	52.76%	57.95%	52.11%	46.14%	58.21%	51.55%	38.64%
DANet	52.03%	59.48%	51.59%	48.52%	56.16%	51.48%	34.63%
DANet+UG	55.55%	62.09%	56.24%	51.90%	57.89%	49.96%	38.55%
DANet+USIM	56.05%	58.07%	58.36%	49.72%	50.30%	49.62%	42.28%
DeepLabV3+	46.88%	63.76%	60.70%	43.64%	55.39%	46.07%	35.27%
DeepLabV3++UG	48.91%	60.64%	58.75%	44.25%	49.53%	52.38%	37.02%
DeepLabV3++USIM	50.79%	58.29%	55.10%	49.74%	45.19%	51.00%	35.84%
DLR9	53.37%	59.59%	56.22%	50.36%	49.85%	53.21%	35.56%
DLR9+UG	61.18%	61.88%	55.43%	51.68%	60.90%	62.09%	52.32%
DLR9+USIM	54.20%	68.67%	54.21%	47.20%	49.29%	57.64%	41.93%
ESPCN+TreeUNet	49.90%	62.84%	62.09%	51.62%	50.85%	51.54%	46.93%

FCN	51.36%	65.36%	57.89%	46.64%	50.78%	51.74%	32.42%
FCN+UG	51.97%	66.76%	57.51%	42.46%	47.93%	48.21%	40.48%
FCN+USIM	48.09%	62.97%	60.47%	44.56%	47.27%	53.65%	36.56%
FusionNet	48.73%	65.53%	53.88%	42.66%	50.55%	44.57%	37.67%
FusionNet+UG	53.97%	57.90%	55.97%	45.67%	50.10%	49.36%	37.00%
FusionNet+USIM	55.87%	59.52%	59.01%	43.85%	48.31%	46.24%	36.99%
Red-Net	53.44%	67.58%	56.54%	51.53%	51.02%	57.98%	45.10%
Red-Net+UG	59.26%	67.17%	59.39%	54.05%	52.13%	51.59%	53.14%
Red-Net+USIM	51.20%	61.53%	54.66%	46.40%	49.88%	51.57%	42.00%
RPCNet	57.88%	58.64%	61.60%	49.95%	52.20%	51.68%	41.31%
RPCNet+UG	57.90%	59.82%	59.47%	49.32%	56.33%	61.06%	48.66%
RPCNet+USIM	51.61%	58.96%	56.19%	48.83%	48.95%	49.16%	43.72%
TreeUNet	56.64%	65.79%	57.90%	46.93%	49.60%	55.12%	44.05%
TreeUNet+UG	58.30%	62.16%	58.09%	56.84%	54.50%	62.71%	50.04%
TreeUNet+USIM	51.79%	62.89%	60.23%	54.32%	49.92%	49.58%	47.18%
U-Net	49.27%	64.52%	56.88%	47.30%	50.65%	52.85%	37.25%
U-Net+UG	56.35%	62.04%	59.05%	52.03%	54.08%	48.58%	40.70%
U-Net+USIM	52.61%	65.94%	51.88%	44.62%	54.58%	47.02%	37.77%
U-NetPPL	51.62%	67.09%	55.48%	51.95%	50.89%	50.51%	39.43%
U-NetPPL+UG	55.88%	65.68%	60.11%	48.20%	56.30%	53.64%	39.54%
U-NetPPL+USIM	53.27%	63.19%	60.44%	43.81%	50.55%	52.12%	38.64%

Appendix B-3-2. Quantitative analysis

Figure 10: Segmentation results on **KUD** dataset

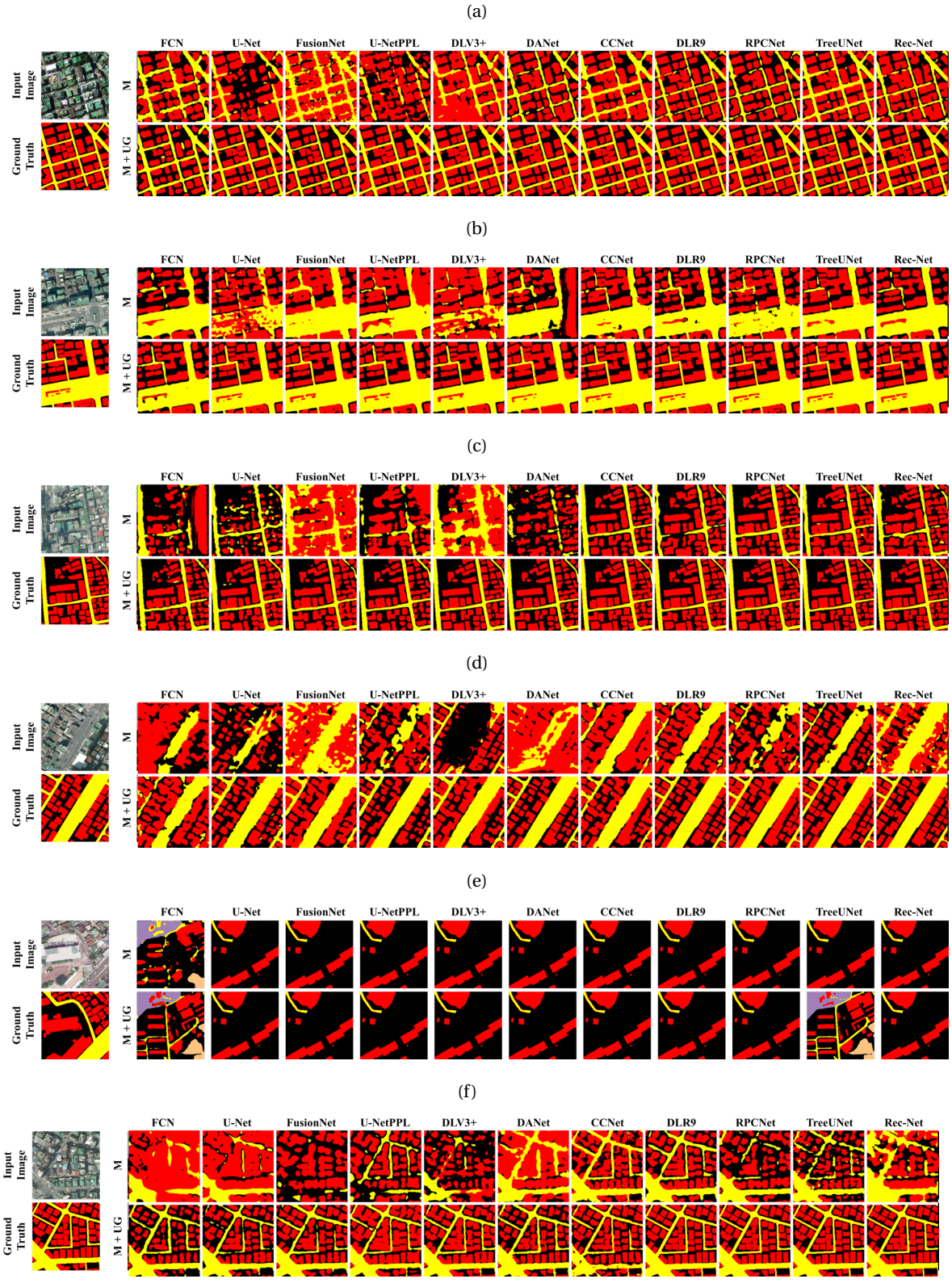
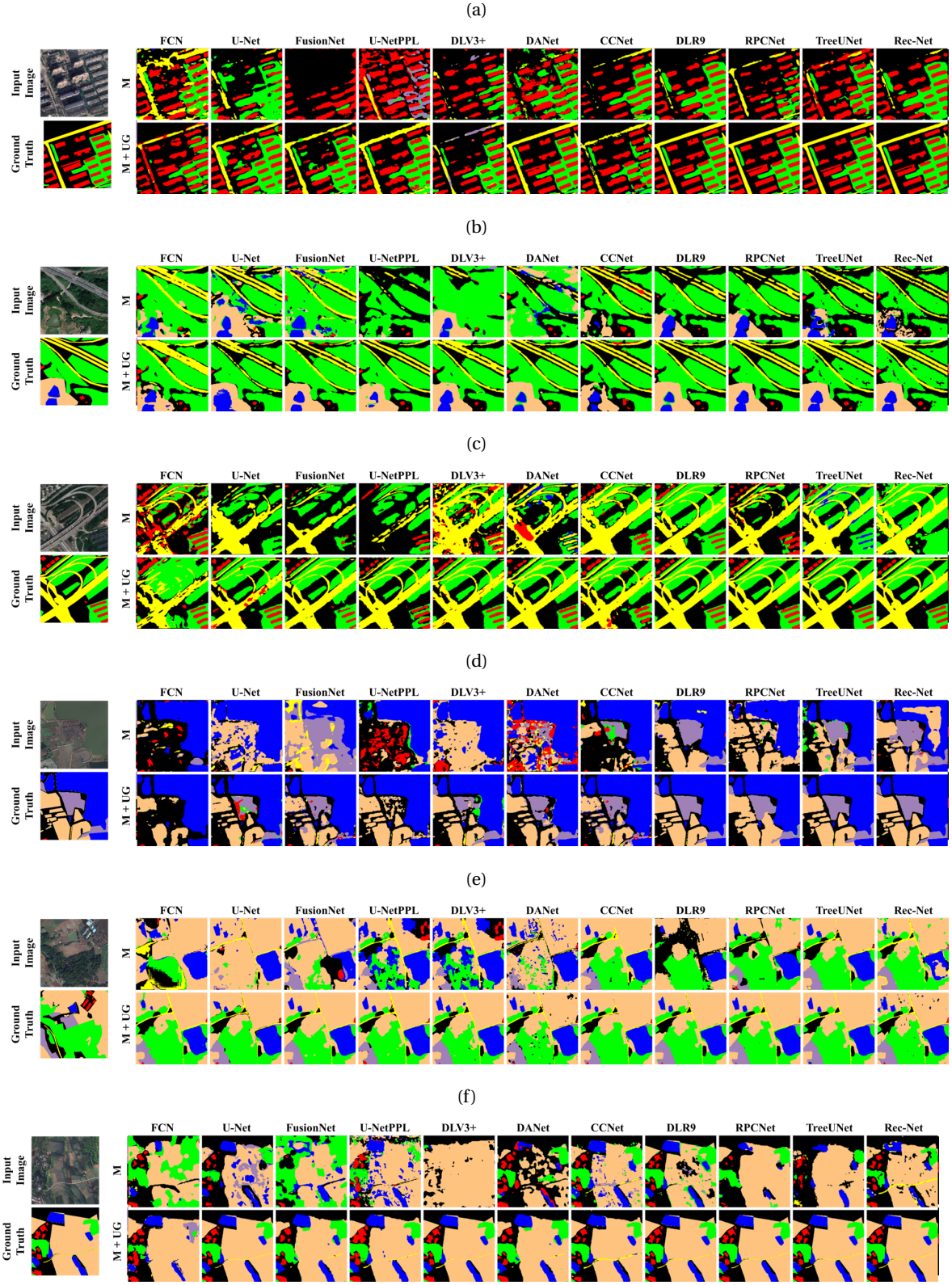


Figure 11: Segmentation results on **LoveDA** dataset



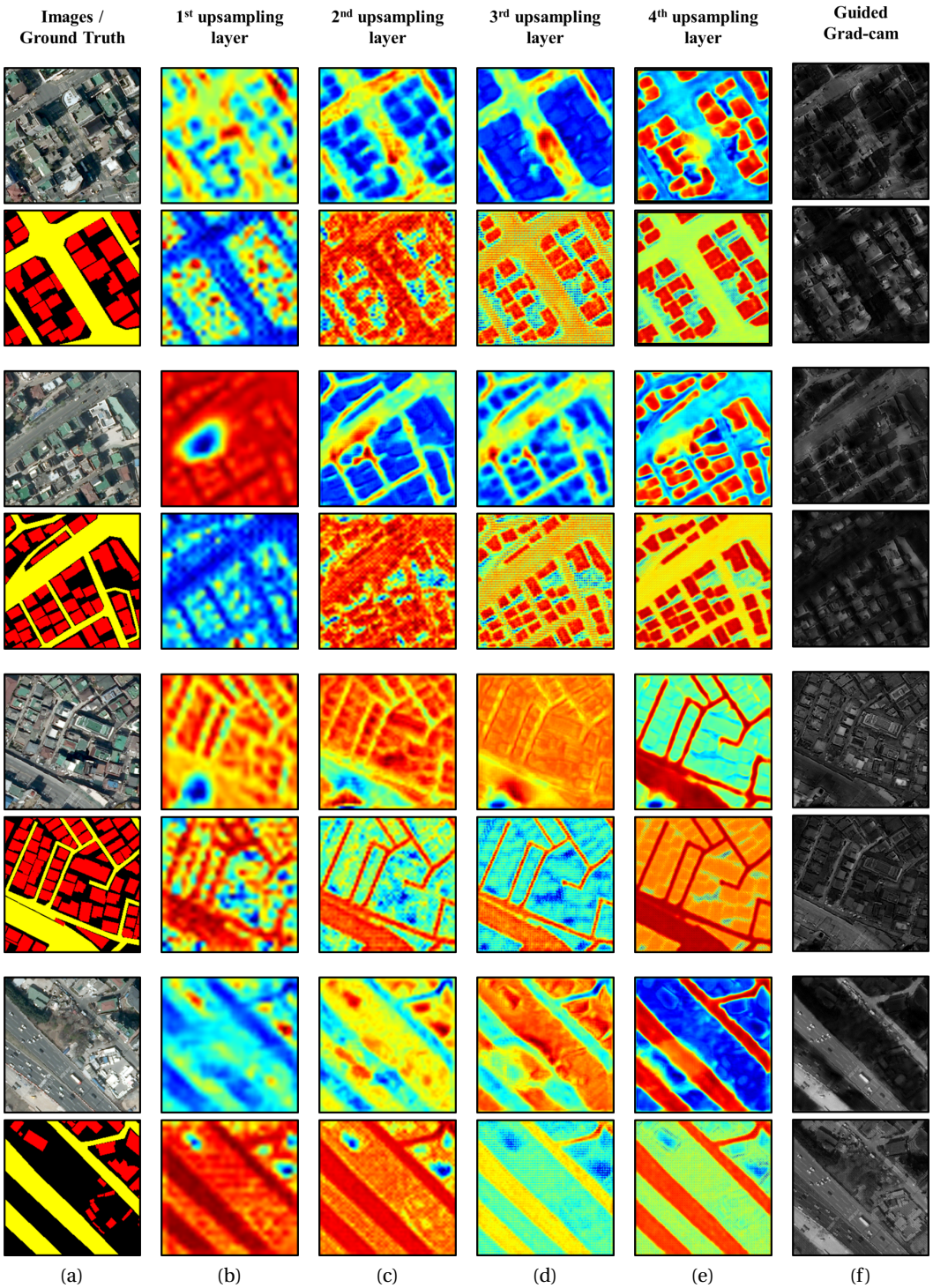


Figure 12: (a) Input image and the corresponding ground truth for buildings (yellow) and roads (red). (b) Grad-CAM of 1st-upsampling layer with deconvolution and concatenation (up) and USIM Gate (down). (c) Grad-CAM of 2nd-upsampling layer with deconvolution and concatenation (up) and USIM Gate (down). (d) Grad-CAM of 3rd-upsampling layer with deconvolution and concatenation (up) and USIM Gate (down). (e) Grad-CAM of 4th-upsampling layer with deconvolution and concatenation (up) and USIM Gate (down). (f) The guided grad-cam with deconvolution and concatenation (up) and USIM Gate (down). First two sets targeted building class and other two targeted roads class.