remote sensing

Article

Refined UNet V4: End-to-End Patch-Wise Network for Cloud and Shadow Segmentation with Bilateral Grid

Libin Jiao [1], Lianzhi Huo [2], Changmiao Hu [2], Ping Tang [2] and Zheng Zhang [2,*]

[1] Department of Computer Science and Technology, School of Mechanical Electronic and Information Engineering, China University of Mining and Technology-Beijing, Beijing 100083, China; jiaolibin@cumtb.edu.cn

[2] Aerospace Information Research Institute (AIR), Chinese Academy of Sciences (CAS), Beijing 100101, China; huolz@aircas.ac.cn (L.H.); hucm@aircas.ac.cn (C.H.); tangping@aircas.ac.cn (P.T.)

* Correspondence: zhangzheng@aircas.ac.cn

Abstract: Remote sensing images are usually contaminated by cloud and corresponding shadow regions, making cloud and shadow detection one of the essential prerequisites for processing and translation of remote sensing images. Edge-precise cloud and shadow segmentation remains challenging due to the inherent high-level semantic acquisition of current neural segmentation fashions. We, therefore, introduce the Refined UNet series to partially achieve edge-precise cloud and shadow detection, including two-stage Refined UNet, v2 with a potentially efficient gray-scale guided Gaussian filter-based CRF, and v3 with an efficient multi-channel guided Gaussian filter-based CRF. However, it is visually demonstrated that the locally linear kernel used in v2 and v3 is not sufficiently sensitive to potential edges in comparison with Refined UNet. Accordingly, we turn back to the investigation of an end-to-end UNet-CRF architecture with a Gaussian-form bilateral kernel and its relatively efficient approximation. In this paper, we present Refined UNet v4, an end-to-end edge-precise segmentation network for cloud and shadow detection, which is capable of retrieving regions of interest with relatively tight edges and potential shadow regions with ambiguous edges. Specifically, we inherit the UNet-CRF architecture exploited in the Refined UNet series, which concatenates a UNet backbone of coarsely locating cloud and shadow regions and an embedded CRF layer of refining edges. In particular, the bilateral grid-based approximation to the Gaussian-form bilateral kernel is applied to the bilateral message-passing step, in order to ensure the delineation of sufficiently tight edges and the retrieval of shadow regions with ambiguous edges. Our TensorFlow implementation of the bilateral approximation is relatively computationally efficient in comparison with Refined UNet, attributed to the straightforward GPU acceleration. Extensive experiments on Landsat 8 OLI dataset illustrate that our v4 can achieve edge-precise cloud and shadow segmentation and improve the retrieval of shadow regions, and also confirm its computational efficiency.

Keywords: edge-precise cloud and shadow segmentation; end-to-end segmentation solution; CRF-based edge refinement

check for updates

Citation: Jiao, L.; Huo, L.; Hu, C.; Tang, P.; Zhang, Z. Refined UNet V4: End-to-End Patch-Wise Network for Cloud and Shadow Segmentation with Bilateral Grid. *Remote Sens.* **2022**, *14*, 358. https://doi.org/10.3390/rs14020358

Academic Editor: Józef Lisowski

Received: 9 December 2021
Accepted: 7 January 2022
Published: 13 January 2022

1. Introduction

Remote sensing images are usually contaminated by cloud and corresponding shadow regions when acquired, which notoriously perturbs the recognition of land cover and finally leads to invalid resolved results [1]; furthermore, more and more remote sensing applications require cloud- and shadow-free images [2–6]. Cloud and shadow detection, therefore, is one of the essential prerequisites for processing and translation of remote sensing images [7]. Since each pixel within a remote sensing image should be identified as the category of cloud, shadow, or background, intelligent cloud and shadow detection is in practice formulated as a semantic segmentation task, driven by large-scale coarsely labeled data and sophisticated neural segmentation models.

In practice, sophisticated neural semantic segmentation fashions have been developed as powerful network backbones for representative feature extraction are arising from well-customized classification networks. Since the Fully-connected Convolutional Network (FCN) [8] initiated neural semantic segmentation by reusing a well-trained backbone of a classification network and concatenating a specified head layer for pixel-wise classification, its successors have achieved terrific quantitative and qualitative performance in various natural image segmentation tasks. Growing neural segmentation models can also enable pixel-wise differentiation in other specified domains.

Edge-precise cloud and shadow segmentation, however, remains challenging due to the inherent high-level semantic acquisition of current neural segmentation fashions: receptive fields are growing rapidly to capture more comprehensive global features, or longer-range features are combined by the cutting-edge models, which contribute more to sensing high-level semantic information rather than low-level vision features. It is accordingly observed that state-of-the-art models yield semantically accurate coverages but cannot produce sufficiently edge-precise segmentation proposals. Edge-sensitive approaches refine the segmentation performance around edges in a pipeline way, using nonlinear filters to visually improve segmentation proposals. As a consequence, it is worth developing an end-to-end solution to the edge-precise cloud and shadow segmentation.

We introduce a UNet-CRF architecture to address the issue of edge-precise cloud and shadow detection, the instances of which are referred to as the Refined UNet series [1,2,7]. Refined UNet [1] is a pipeline of UNet backbone together with an offline Dense CRF post-processing to coarsely locate cloud and shadow regions and refine their edges; Refined UNet v2 [2] and v3 [7] are end-to-end segmentation solutions that, respectively, introduce gray-scale and multi-channel guided Gaussian filters to facilitate the efficient bilateral message-passing step, which get closer to our ultimate goal of discovering effective and efficient end-to-end edge-precise segmentation solutions. However, it is visually demonstrated that the locally linear kernel used in v2 and v3 is not sufficiently sensitive to potential edges, in comparison with Refined UNet with the Gaussian-form bilateral kernel. We, therefore, turn back to the investigation of an end-to-end UNet-CRF architecture with a Gaussian-form bilateral kernel and its relatively efficient approximation. In this paper, we present Refined UNet v4, an end-to-end edge-precise segmentation network for cloud and shadow detection, which is capable of retrieving regions of interest with relatively tight edges and potential shadow regions with ambiguous edges. Specifically, we inherit the UNet-CRF architecture exploited in the Refined UNet series, which concatenates a UNet backbone of coarsely locating cloud and shadow regions and an embedded CRF layer of refining edges. In particular, the bilateral grid-based approximation to the Gaussian-form bilateral kernel is applied to the bilateral message-passing step, in order to ensure the delineation of sufficiently tight edges and the retrieval of shadow regions with ambiguous edges. Our TensorFlow implementation of the bilateral approximation is relatively computationally efficient in comparison with Refined UNet, attributed to the straightforward GPU acceleration. An illustration of qualitative differences between the Refined UNet series is shown in Figure 1. Consequently, the main contributions of this paper as listed as follows.

- Refined UNet v4: we propose an end-to-end network for cloud and shadow segmentation of remote sensing images, which can perform cloud and shadow detection in an edge-precise way, improve the retrieval of shadow regions with potential edges, and also enable a relatively speed-up in comparison with Refined UNet [1].
- Bilateral grid-based relatively efficient CRF inference: the bilateral grid-based message-passing kernel is introduced to form the bilateral step in CRF inference, and it is demonstrated that the bilateral step can be straightforwardly characterized by the sophisticated implementations of the bilateral filter.
- Generalization to the RICE dataset: we generalize our v4 to the RICE dataset. The experiment shows that our v4 can also perform edge-precise detection of regions of interest.

- Open access of Refined UNet v4: A pure TensorFlow implementation is given and publicly available at https://github.com/92xianshen/refined-unet-v4 (accessed on 22 December 2021).

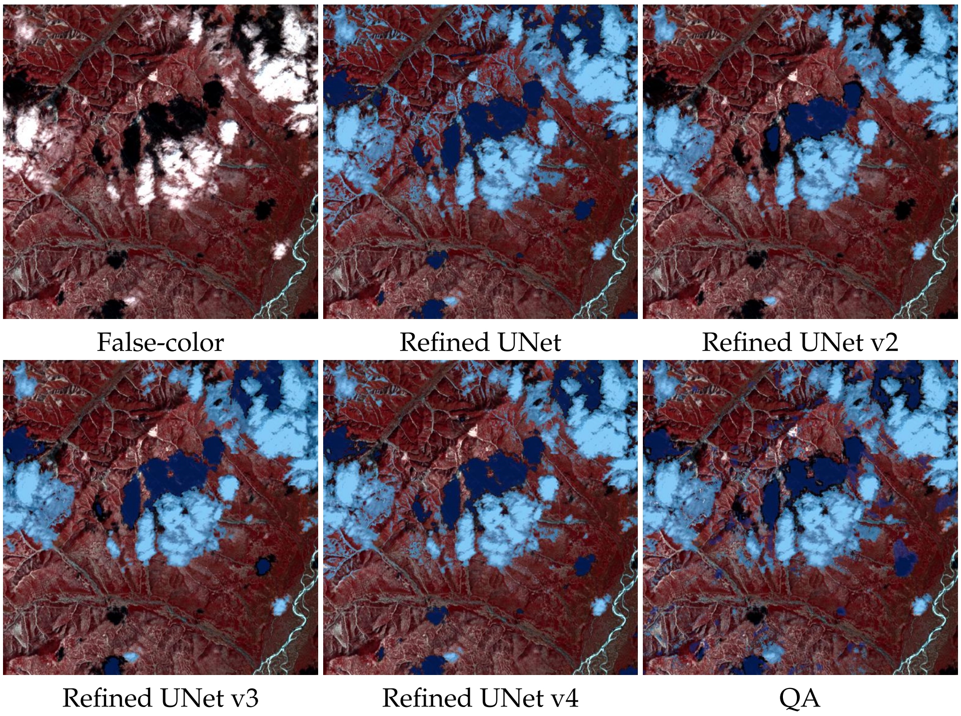

False-color
Refined UNet
Refined UNet v2
Refined UNet v3
Refined UNet v4
QA

Figure 1. Illustration of qualitative differences between the Refined UNet series [1,2,7]. False-color and QA denote the false-color (Bands 5 NIR, 4 Red, and 3 Green) and label reference images. Refined UNet v4 is visually comparable to Refined UNet, in terms of edge-precise segmentation of cloud and shadow regions. In addition, v4 is able to retrieve more potential shadow regions with ambiguous edges, in comparison with v2 and v3.

The rest of the paper is organized as follows. Section 2 recaps related work regarding neural network segmentation, CRF-based refinement, and sophisticated implementations of the bilateral filter. Section 3 presents the overall framework and the distinct contribution of our Refined UNet v4. Section 4 introduces the experimental setups, the quantitative, and the visual evaluations of the presented methods, Section 5 concludes the paper.

2. Related Work

We recap related literature regarding neural image segmentation, corresponding refinement techniques, and efficient solutions to edge-preserving filters, which summarizes innovative segmentation techniques for our edge-precise cloud and shadow differentiation.

2.1. Neural Semantic Segmentation Revisited

Neural semantic segmentation was initiated by Fully Convolutional Neural Networks (FCN) [8] that exploited a well-trained convolutional module as its feature backbone and yielded dense predictions at its customized head layer. More and more pretrained neural backbone-based segmentation models thrive since then, in terms of their significant improvement on the quantitative scores and visual results. The upstream vision task, known as image classification, provides fundamental structures of neural networks, and sufficient image sets, ImageNet [9] for example, act as benchmarks to obtain well-trained parameters as well. Typical neural image classifiers arise from the image classification tasks, such as VGG-16/VGG-19 [10], MobileNets V1/V2/V3 [11–13], ResNet18/ResNet50/ResNet101 [14,15], DenseNet [16], and EfficientNet [17], and their convolutional modules have been applied to well-designed segmentation models.

It is noted that neural segmentation techniques customize its feature module to comprehend high-level semantic information, including approaches of enlarging receptive fields [18–22] and of concatenating dilated convolutions with various rates [23–28]. On the other hand, they also introduce particular fashions to preserve low-level vision features, including feeding multi-scale input images [19,29,30], fusing intermediate feature maps [31,32], and building skip connections [14,15,18]. In addition, encoder-decoder architectures [18,22,33,34] provide a structural solution to the dense prediction. Typical segmentation models, such as U-Net [18], RefineNet [29], PSPNet [31], FastFCN [35], and DeepLab series [23–26,36], achieve promising segmentation performance and will be possibly extended to other related domains. In addition, a lightweight panoramic annular neural net [37] and a sparse point-wise affinity map [38] were introduced to perform aerial image segmentation. Recently, transformer-based segmentation models [39–45] are arising because of their long-range feature extraction. The aforementioned techniques have a great inspiration for extending neural segmentation to other related domains. However, these techniques focus more on long-range semantic information, giving rise to significant improvement in coarse-grained instance segmentation. The transformer-based methods, in particular, are capable of capturing global semantic information to obtain segmentation gain at the instance level. It is speculated that their property of long-range semantic perception, nevertheless, may discourage their acquisition of low-level visual features and fine-grained edge refinement, illustrated by their typical results. We will discuss the effect of long-range perception in our future work but restrict our attention to the effect of low-level visual features, and we select lightweight but effective UNet [18] as the segmentation backbone.

2.2. Segmentation Refinement Revisited

Sophisticated segmentation models are able to capture high-level semantic information as well as low-level vision features, which is not sufficiently compatible. Concurrent refinement approaches can basically be grouped into online and offline categories, in terms of performing within the forward propagation of the model or as post-processing. Typical online techniques refine low-level visual performance by manipulating the scales of receptive fields [18–22], recycling features from the frontend [18,29,31,33,34], or introducing gradient discrepancy [46] to constrain. An exemplar of offline techniques is CRF post-processing [23,24,30,47–51]. Besides, guided filter [52] is adopted to improve visual performance. Innovative architectures, such as CRFasRNN [49] and learnable guided filter [53], applied particular module to refine the segmentation proposals. These techniques offer edge-precise fashions in image segmentation tasks. We are motivated to customize CRF to fit our one-stage edge-precise solution.

2.3. Efficient Solutions to Edge-Preserving Filters

Dense CRF [47,48] expressed the bilateral message-passing step as a Gaussian-form convolution and applied permutohedral lattice-based convolution to its efficient solution. We are motivated to delve into its insight of the high-dimensional Gaussian filter-based bilateral message-passing step and then collect relevant techniques to fit our edge-precise detection. Approximate computation for Gaussian-form bilateral filter significantly speeds up the nonlinear filtering, using Taylor expansion [54], trigonometric range kernel approximation [55], iteration of square window-based filter [56], memory- and computation-efficient iteration [57], linearization with fast Fourier transformation [58], and fast high-dimensional filter [59–61], respectively. In particular, the bilateral grid [59,62,63] facilitates the implementation of efficient approximate computation, which is currently applied to our Gaussian-form bilateral message-passing step.

3. Refined UNet V4 for Edge-Precise Segmentation

We present the overall framework of our Refined UNet v4 in this section, which comprises the UNet backbone, embedded CRF refinement layer, and the implementation of the bilateral grid-based bilateral message-passing step.

3.1. UNet Prediction and Conditional Random Field-Based Refinement Revisited

We first revisit the feedforward propagation of our UNet-CRF architecture for cloud and shadow segmentation: the proposed architecture takes as input a local patch of a seven-band high-resolution image and outputs two sorts of segmentation proposals: a coarsely labeled proposal and an edge-refined one as well. Note that the ultimate goal of edge-precise segmentation is to predict and refine the segmentation proposal in a one-stage way, instead of a pipeline comprising an online coarse localization and an offline post-process; this accounts for our intention started from Refined UNet v2 [2]. In particular, we extract local patches and restore full segmentation proposals from patches in an offline way, the improvement of which will be discussed in our future work. The overall framework is illustrated in Figure 2.

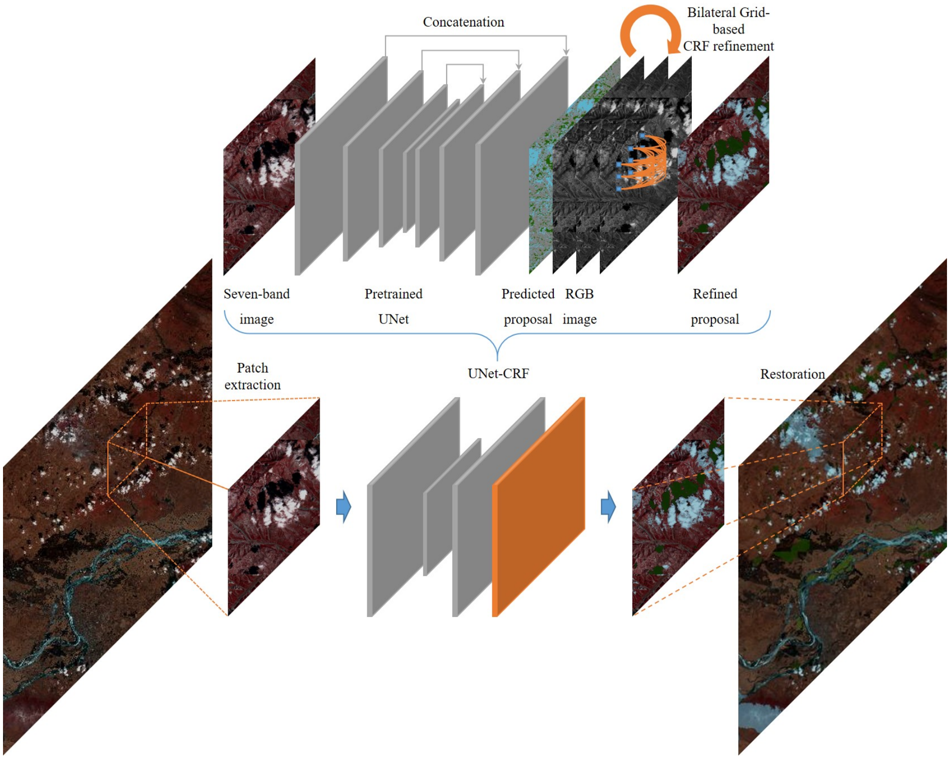


Figure 2. Illustration of the overall framework of Refined UNet v4. The UNet-CRF architecture copes with local patches extracted from original full images: the pretrained UNet yields predicted segmentation proposals while the embedded CRF layer refines the edges. Full refined proposals are finally restored from local patches. We make use of the bilateral grid to facilitate the nonlinear bilateral message-passing step, which is equivalent to the Gaussian-form step in Dense CRF [47].

We now turn to the specific implementation of UNet-CRF concatenating a UNet backbone for coarse localization and an embedded CRF layer for edge refinement. UNet [18] has been proven as a sophisticated model for semantic segmentation in computer vision tasks, which contains four down-sampling blocks of “Convolution-ReLU-MaxPooling” and four up-sampling blocks of “UpSampling-Convolution-ReLU”. In particular, residual connections bridge intermediate feature maps with the same resolutions, which is considered as the feature reuse and shortcuts of gradient backpropagation. UNet finally yields the categorical likelihood tensor at its head layer and returns the class of each pixel by the indices of the maximum values along the categorical dimension. Please note that this architecture has been thoroughly discussed in Refined UNet [1], v2 [2], and v3 [7]; we, therefore, briefly recap the structure of UNet and do not provide in detail any longer.

We also briefly revisit the CRF-based spatial refinement for cloud and shadow segmentation. We formulate our dense classification as a conditional random field (CRF) characterized by Gibbs distribution [47], given the multi-channel global observation $\boldsymbol{I}$. The formulation is given by

$$P(\boldsymbol{X}|\boldsymbol{I}) = \frac{1}{Z(\boldsymbol{I})}\exp(-E(\boldsymbol{x}|\boldsymbol{I})) \tag{1}$$

in which $E(\boldsymbol{x})$ denotes the Gibbs energy, $\boldsymbol{X}$ is the element-wise classification, and $Z(\boldsymbol{I})$ is the normalization factor.

The maximum a posteriori (MAP) element-wise classification $\boldsymbol{x}^*$ arises when the probability $P(\boldsymbol{x}|\boldsymbol{I})$ reaches the maximum value, given by

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\mathcal{L}^N} P(\boldsymbol{x}|\boldsymbol{I}). \tag{2}$$

Specifically, the Gibbs energy $E(\boldsymbol{x}|\boldsymbol{I})$ has the form of

$$E(\boldsymbol{x}|\boldsymbol{I}) = \sum_i \psi_u(x_i) + \sum_i \sum_{j<i} \psi_p(x_i, x_j) \tag{3}$$

in which $\psi_u(x_i)$ denotes the unary potential generated from the spatially coarse prediction of the UNet backbone, and $\psi_p(x_i, x_j)$ is the pairwise potential describing the relevance of adjacent x_i and x_j.

In particular, the pairwise potential in our case has the form of

$$\psi_p(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^{K} w^{(m)} k^{(m)}(\boldsymbol{f}_i, \boldsymbol{f}_j) \tag{4}$$

in which $\mu(x_i, x_j)$ is the label compatibility function, $w^{(m)}$ is the weight of the m^{th} message-passing kernel, and $k^{(m)}$ is the m^{th} Gaussian-form message-passing kernel.

In our case, Potts model [47] is employed as the label compatibility function $\mu(x_i, x_j)$, having the form of $\mu(x_i, x_j) = [x_i \neq x_j]$. Feature vectors $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ comprise either spatial positions $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$, or both spatial positions and color intensities $\boldsymbol{p}_i$, $\boldsymbol{I}_i$, $\boldsymbol{p}_j$, and $\boldsymbol{I}_j$, giving rise to the form of

$$k(\boldsymbol{f}_i, \boldsymbol{f}_j) = w^{(1)} \exp\left(-\frac{||\boldsymbol{p}_i - \boldsymbol{p}_j||^2}{2\theta_\alpha^2} - \frac{||\boldsymbol{I}_i - \boldsymbol{I}_j||^2}{2\theta_\beta^2}\right) + w^{(2)} \exp\left(-\frac{||\boldsymbol{p}_i - \boldsymbol{p}_j||^2}{2\theta_\gamma^2}\right). \tag{5}$$

We refer to $k^{(2)}(\boldsymbol{f}_i, \boldsymbol{f}_j)$ as the spatial message-passing kernel, which is only related to spatial positions $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$, given by

$$k^{(2)}(\boldsymbol{f}_i, \boldsymbol{f}_j) = \exp\left(-\frac{||\boldsymbol{p}_i - \boldsymbol{p}_j||^2}{2\theta_\gamma^2}\right). \tag{6}$$

On the other hand, $k^{(1)}(\boldsymbol{f}_i, \boldsymbol{f}_j)$ is referred to as the bilateral message-passing step with respect to both spatial positions $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ and color intensities $\boldsymbol{I}_i$ and $\boldsymbol{I}_j$, which will be thoroughly discussed in the next subsection.

The mean-field approximation [47] yields an approximated value of $P(\boldsymbol{X}|\boldsymbol{I})$ in an efficient way, instead of computing an exact MAP result. The iterative update method thus has such a form of

$$\boldsymbol{Q}_i = \frac{1}{Z_i} \exp\left(-\boldsymbol{\psi}_u(x_i) - \boldsymbol{\mu} \sum_{m=1}^{K} w^{(m)} \sum_{j \neq i} k^{(m)}(\boldsymbol{f}_i, \boldsymbol{f}_j) \boldsymbol{Q}_j \right). \tag{7}$$

Furthermore, we finally have the potentially possible label assignment when the update converged.

We have presented the overall framework of our proposed UNet-CRF segmentation architecture, except for the implementation of the bilateral message-passing step. Actually, we have comprehensively investigated multiple implementations of the bilateral step, such as high-dimensional Gaussian filter-based offline Dense CRF in Refined UNet [1], the grayscale guided Gaussian filter-based CRF in Refined UNet v2 [2], and the multi-channel guided Gaussian filter-based CRF in Refined UNet v3 [7]. In this work, we turn back to the exploration of the Gaussian-form bilateral kernel and attempt to reveal the potentially significant difference between Gaussian-form and locally linear kernels.

3.2. Bilateral Grid-Based Bilateral Message-Passing Step

We present in detail the bilateral grid-based approximation to Gaussian-form bilateral message-passing step. First, we clarify the notations of the bilateral step. A flattened global observation (remote sensing image) comprises HW color intensity vectors $\boldsymbol{I}_i$:

$$\{\boldsymbol{I}_1, \boldsymbol{I}_2, \ldots, \boldsymbol{I}_{HW}\} \in \mathbb{R}^{HW \times C}. \tag{8}$$

In our case, $\boldsymbol{I}_i$ contains three channels, including R, G, and B ($C = 3$):

$$\boldsymbol{I}_i = \left[I_{i,r}, I_{i,g}, I_{i,b} \right]^\top \in \mathbb{R}^C. \tag{9}$$

Similarly, flattened $\boldsymbol{Q}$ comprises HW likelihood vectors $\boldsymbol{Q}_i$:

$$\{\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_{HW}\} \in \mathbb{R}^{HW \times N}. \tag{10}$$

But $\boldsymbol{Q}_i$ is the N-dimensional vector denoting N-class classification, given by

$$\boldsymbol{Q}_i = [Q_{i,0}, Q_{i,1}, \ldots, Q_{i,N-1}]^\top \in \mathbb{R}^N. \tag{11}$$

We currently turn to the introduction to our bilateral approximation. For simplification, we extract the bilateral message-passing step, given by

$$\tilde{\boldsymbol{Q}}_i^{(1)} = \sum_{j \neq i} k^{(1)}(\boldsymbol{f}_i, \boldsymbol{f}_j) \boldsymbol{Q}_j. \tag{12}$$

And $k^{(1)}(\boldsymbol{f}_i, \boldsymbol{f}_j)$ should have the desirable bilateral form of

$$k^{(1)}(\boldsymbol{f}_i, \boldsymbol{f}_j) = \exp\left(-\frac{||\boldsymbol{p}_i - \boldsymbol{p}_j||^2}{2\theta_\alpha^2} - \frac{||\boldsymbol{I}_i - \boldsymbol{I}_j||^2}{2\theta_\beta^2} \right). \tag{13}$$

To facilitate the nonlinear computation of color fraction, we approximate (12) by the bilateral grid discussed in the fast bilateral filter [59]. Specifically, the additional dimensions $\boldsymbol{\zeta}$ are introduced to map the feature into a high-dimensional representation $(j, \boldsymbol{\zeta})$. A Kronecker indicator δ is also employed to signify a valid transformation, given by

$$
\begin{aligned}
\tilde{\boldsymbol{Q}}_i^{(1)} &= \sum_{j \neq i} \exp\left(-\frac{||\boldsymbol{p}_i - \boldsymbol{p}_j||^2}{2\theta_\alpha^2} - \frac{||\boldsymbol{I}_i - \boldsymbol{I}_j||^2}{2\theta_\beta^2} \right) \boldsymbol{Q}_j \\
&= \sum_{j \neq i} \sum_{\boldsymbol{\zeta} \in \mathcal{R}} \exp\left(-\frac{||\boldsymbol{p}_i - \boldsymbol{p}_j||^2}{2\theta_\alpha^2} - \frac{||\boldsymbol{I}_i - \boldsymbol{\zeta}||^2}{2\theta_\beta^2} \right) \delta(\boldsymbol{\zeta} - \boldsymbol{I}_j) \boldsymbol{Q}_j \\
&= \sum_{\substack{(j,\boldsymbol{\zeta}) \in \mathcal{S} \times \mathcal{R} \\ j \neq i \\ \boldsymbol{\zeta} \neq \boldsymbol{I}_i}} g_{\theta_\alpha, \theta_\beta}(\boldsymbol{p}_i - \boldsymbol{p}_j, \boldsymbol{I}_i - \boldsymbol{\zeta}) \boldsymbol{q}_{j,\boldsymbol{\zeta}}
\end{aligned}
\tag{14}
$$

in which δ is defined by

$$
\delta(\boldsymbol{\zeta}) = \begin{cases} 0, \text{ if } \boldsymbol{\zeta} = \boldsymbol{0} \\ 1, \text{ otherwise} \end{cases}
\tag{15}
$$

and the high-dimensional mapping of $\boldsymbol{Q}_j$ is defined by

$$
\boldsymbol{q}_{j,\boldsymbol{\zeta}} = \delta(\boldsymbol{\zeta} - \boldsymbol{I}_j) \boldsymbol{Q}_j.
\tag{16}
$$

The high-dimensional transformation leads to a linear high-dimensional convolution over $\mathcal{S} \times \mathcal{R}$, denoted by $g_{\theta_\alpha, \theta_\beta}$. Consequently, the high-dimensional transformation, convolution, and inverse transformation can be formulated by a pipeline of *splatting*, *blurring*, and *slicing*, as discussed as follows.

3.2.1. Splat

We redefine the spatial position $\boldsymbol{p}_j$ as $\left[p_{j,h}, p_{j,w} \right]^\top$ and the color intensity $\boldsymbol{I}_j$ as $\left[I_{j,r}, I_{j,g}, I_{j,b} \right]^\top$. We also use $s \in \{h, w\}$ to indicate the spatial axis and $c \in \{r, g, b\}$ to indicate the color axis. We first compute the channel-wise color discrepancy of each pixel by

$$
\Delta I_{j,c} = I_{j,c} - \min(I_{:,c}).
\tag{17}
$$

The high-dimensional coordinate $\boldsymbol{\xi}_j$ of splatting is given by

$$
\boldsymbol{\xi}_j = \left[\xi_{j,h}, \xi_{j,w}, \xi_{j,r}, \xi_{j,g}, \xi_{j,b} \right]^\top
\tag{18}
$$

in which

$$
\xi_{j,s} = \left\lfloor \frac{p_{j,s}}{\theta_\alpha} + \frac{1}{2} \right\rfloor + \epsilon_s
\tag{19}
$$

$$
\xi_{j,c} = \left\lfloor \frac{\Delta I_{j,c}}{\theta_\beta} + \frac{1}{2} \right\rfloor + \epsilon_c.
\tag{20}
$$

For each $Q_{j,n} \in \boldsymbol{Q}_{:,n}$, $q_{\boldsymbol{\xi}_j}$ computes a cumulative summation, given by

$$
q_{\boldsymbol{\xi}_j} \leftarrow q_{\boldsymbol{\xi}_j} + Q_{j,n}.
\tag{21}
$$

3.2.2. Blur

For each axis $k \in \{h, w, r, g, b\}$, we perform a spatial convolution, given by

$$
q_{\xi_{j,k}} \leftarrow \frac{1}{2} q_{\xi_{j-1,k}} + \frac{1}{2} q_{\xi_{j+1,k}}.
\tag{22}
$$

The convolution should be repeated M times.

3.2.3. Slice

Slicing is actually a multi-dimensional linear interpolation in our case. For the jth element, we first compute its spatially previous and next coordinates $\xi^{-}_{j,s}$ and $\xi^{+}_{j,s}$, given by

$$\xi^{-}_{j,s} = \max\left(0, \min\left(L_s - 1, \left\lfloor \frac{p_{j,s}}{\theta_\alpha} + \epsilon_s \right\rfloor\right)\right) \tag{23}$$

and

$$\xi^{+}_{j,s} = \max\left(0, \min\left(L_s - 1, \xi^{-}_{j,s} + 1\right)\right), \tag{24}$$

respectively, in which L_s denotes the spatial range (height H or width W).

The spatial interpolation factor is also given by

$$\alpha_{j,s} = \frac{p_{j,s}}{\theta_\alpha} + \epsilon_s - \xi^{-}_{j,s}. \tag{25}$$

Similarly, the color range (depth of each color channel) is given by

$$D_c = \left\lfloor \frac{\max(I_{:,c}) - \min(I_{:,c})}{\theta_\beta} \right\rfloor + 1 + 2\epsilon, \tag{26}$$

and the previous and next color coordinates are given by

$$\xi^{-}_{j,c} = \max\left(0, \min\left(D_c - 1, \frac{\Delta I_{j,c}}{\theta_\beta} + \epsilon_c\right)\right) \tag{27}$$

and

$$\xi^{+}_{j,c} = \max\left(0, \min\left(D_c - 1, \xi^{-}_{j,c} + 1\right)\right). \tag{28}$$

The color interpolation factor is also given by

$$\alpha_{j,c} = \frac{\Delta I_{j,c}}{\theta_\beta} + \epsilon_c - \xi^{-}_{j,c}. \tag{29}$$

The $\tilde{Q}^{(1)}_{i,n}$ is computed by the multi-dimensional linear interpolation, given by

$$\begin{array}{rllllll}
\tilde{Q}^{(1)}_{i,n} = & (1-\alpha_{j,h}) & \cdot(1-\alpha_{j,w}) & \cdot(1-\alpha_{j,r}) & \cdot(1-\alpha_{j,g}) & \cdot(1-\alpha_{j,b}) & \cdot q_{\xi^{-}_{j,h},\xi^{-}_{j,w},\xi^{-}_{j,r},\xi^{-}_{j,g},\xi^{-}_{j,b}} \\
+ & (1-\alpha_{j,h}) & \cdot(1-\alpha_{j,w}) & \cdot(1-\alpha_{j,r}) & \cdot(1-\alpha_{j,g}) & \cdot\alpha_{j,b} & \cdot q_{\xi^{-}_{j,h},\xi^{-}_{j,w},\xi^{-}_{j,r},\xi^{-}_{j,g},\xi^{+}_{j,b}} \\
+ & (1-\alpha_{j,h}) & \cdot(1-\alpha_{j,w}) & \cdot(1-\alpha_{j,r}) & \cdot\alpha_{j,g} & \cdot(1-\alpha_{j,b}) & \cdot q_{\xi^{-}_{j,h},\xi^{-}_{j,w},\xi^{-}_{j,r},\xi^{+}_{j,g},\xi^{-}_{j,b}} \cdot \\
\dots & & & & & & \\
+ & \alpha_{j,h} & \cdot\alpha_{j,w} & \cdot\alpha_{j,r} & \cdot\alpha_{j,g} & \cdot\alpha_{j,b} & \cdot q_{\xi^{+}_{j,h},\xi^{+}_{j,w},\xi^{+}_{j,r},\xi^{+}_{j,g},\xi^{+}_{j,b}}
\end{array} \tag{30}$$

4. Experiments and Discussion

We experiment with our Refined UNet v4 in the following subsections, including globally quantitative and locally visual comparisons against other methods, hyperparameter test with respect to θ_α and θ_β, ablation study with respect to our bilateral approximation, its computational efficiency, and its generalization to the RICE dataset.

4.1. Experimental Setups, Image Preprocessing, Implementation Details, and Evaluation Metrics Revisited

We first revisit the setups of the experimental dataset inherited from [1]. Specifically, 11 seven-band high-resolution remote sensing images drawn from the Landsat 8 OLI dataset are employed to assess the involved methods, which are acquired in Year 2016 at Path 113 and Row 26, listed as follows.

- 2016-03-27, 2016-04-12, 2016-04-28, 2016-05-14, 2016-05-30, 2016-06-15, 2016-07-17, 2016-08-02, 2016-08-18, 2016-10-21, and 2016-11-06

We make use of Band QA as the label reference, derived from the Level-2 Pixel Quality Assessment band. Numerical class IDs of background, fill value (-9999), shadow, and cloud pixels are assigned to 0, 1, 2, and 3, respectively. In addition, it should be noted that QA is referred to as the label reference instead of the ground truth because the labels of cloud and shadow regions are graphically dilated such that they are not sufficiently precise at the pixel level. The pipeline of image preprocessing is listed as follows: pixels of negative values are assigned to zero, padded full images are sliced into patches of shape 512×512, and each patch is normalized to the interval $[0, 1)$.

For fair comparisons in the experiments, we inherit the pretrained UNet backbone from [1]. The pretrained model has been sufficiently trained and validated, and consequently, the training and validation sets are not provided any longer. The embedded CRF layer is installed as follows. It is built with the TensorFlow [64] framework. We make use of the transformation introduced by [47] to generate unary potentials. The numbers of iterations of mean-field approximation and blurring of the bilateral step are 10 and 2, respectively. θ_α and θ_β are crucial hyperparameters determining the performance of edge-precise detection, and therefore the hyperparameter test is conducted to evaluate the effect of θ_α and θ_β in which θ_α and θ_β vary from 60 to 140 by 20 and 0.0625 to 0.25 by 2, respectively. In particular, the default θ_α and θ_β are empirically assigned to 80 and 0.0625 for both quantitative and visual evaluations, due to their significant performance gain.

We also inherit the assessment metrics from [1]. The quantitative metrics include the overall accuracy (Acc.), Kappa coefficient (κ), mean IoU (mIoU), precision (P), recall (R), and F1 (F_1) scores. Time consumptions are also compared to assess the efficiency of each model instance. Besides, Bands 5 NIR, 4 Red, and 3 Green are merged as RGB channels to construct the false-color visualization.

In the comparative experiments, vanilla UNet reproduced on the training set [1] is used as the baseline method. PSPNet [1,31] is also reproduced as a comparative model instance. The UNet backbone, also referred to as UNet $\times\ \alpha$, and Refined UNets inherited from [1,2,7] are used as comparative methods.

It should be noted that we have inherited plenty of setups of experiments, datasets, and implementations for fair comparisons; in particular, we inherit the test set including 11 high-resolution remote sensing images. Therefore, please kindly refer to [1,2,7] for more details. In addition, we generalize our v4 to the RICE dataset, in order to show our contributions.

4.2. Quantitative Comparisons against Involved Methods

We first compare our v4 against other involved methods from the globally quantitative perspective. We refer to the quantitative comparisons as the global assessment because the indicators evaluate these methods over the entire test set, leading to relatively comprehensive conclusions. However, it should be noted that edge-precise detection is naturally a visual contribution, which can hardly be numerically evaluated. We present numerical evaluation due to verify if our v4 is globally comparable and acceptable. The overall accuracy, Kappa, and mean IoU are overall indicators, and in contrast, the precision, recall, and F1 scores are categorical indicators showing the segmentation performance of each class. The means and the standard deviations of all indicators are presented in Tables 1 and 2, respectively.

Table 1. Average Time, Accuracy, Kappa, and mIoU Scores on Our Landsat 8 OLI Test Set (Mean ± Standard Deviation, $^+$ represents that higher scores indicate better performance).

No.	Models	Time (s/img) [1]	Acc. $^+$ (%)	Kappa $^+$ (%)	mIoU $^+$ (%)
1	**UNet** [1]	-	93.1 ± 6.45	89.06 ± 9.76	65.7 ± 9.38
2	**PSPNet** [1]	-	84.88 ± 7.59	76.51 ± 9.65	53.78 ± 5.97
3	**UNet** $\times \alpha$ [2]	20.67 ± 1.96	93.04 ± 5.45	89.11 ± 7.97	71.94 ± 8.21
4	**Global RFN. UNet** [1]	384.81 ± 5.91	93.48 ± 5.46	89.72 ± 8.12	68.72 ± 7.5
5	**RFN. UNet v2** $(r = 10)$ [2]	61.36 ± 5.25	93.6 ± 5.5	89.93 ± 8.1	71.66 ± 8.14
6	**RFN. UNet v2** $(r = 80)$ [2]	1213.23 ± 4.97	93.38 ± 5.49	89.53 ± 8.16	67.36 ± 7.02
7	**RFN. UNet v3** $(r = 100)$ [7]	82.63 ± 8.32	93.6 ± 5.52	89.9 ± 8.21	69.2 ± 7.6
8	**RFN. UNet v4** $(\theta_\alpha = 60)$ [2]	210.95 ± 14.52	93.63 ± 5.48	89.96 ± 8.14	70.18 ± 7.75
9	**RFN. UNet v4** $(\theta_\alpha = 80)$	207.23 ± 13.89	93.66 ± 5.48	90.0 ± 8.16	69.97 ± 7.76
10	**RFN. UNet v4** $(\theta_\alpha = 100)$	205.24 ± 13.89	93.68 ± 5.46	90.02 ± 8.14	69.79 ± 7.73
11	**RFN. UNet v4** $(\theta_\alpha = 120)$	203.81 ± 13.63	93.68 ± 5.44	90.02 ± 8.13	69.65 ± 7.73
12	**RFN. UNet v4** $(\theta_\alpha = 140)$	199.98 ± 13.48	93.68 ± 5.43	90.02 ± 8.13	69.49 ± 7.77
13	**RFN. UNet v4** $(\theta_\beta = 0.0625)$ [3]	207.23 ± 13.89	93.66 ± 5.48	90.0 ± 8.16	69.97 ± 7.76
14	**RFN. UNet v4** $(\theta_\beta = 0.125)$	202.11 ± 13.16	93.58 ± 5.41	89.85 ± 8.07	68.89 ± 7.31
15	**RFN. UNet v4** $(\theta_\beta = 0.25)$	200.84 ± 13.11	93.5 ± 5.44	89.71 ± 8.12	67.63 ± 7.1

[1] Time consumptions of inference for one full image in the test phase, s/img denotes seconds per image. [2] $\theta_\beta = 0.0625$. [3] $\theta_\alpha = 80$.

As shown in Table 1, no significant differences in the overall accuracy, Kappa, and mean IoU are observed, which demonstrates that our v4 is able to achieve acceptable numerical performance and is also comparable to the involved counterparts. We further turn to the categorical indicators to numerically evaluate the element-wise classification. Please note that the precision and recall scores (P and R) evaluate methods from different perspectives: P indicates the efficacy of correct pixel classification whereas R indicates the efficacy of comprehensive pixel retrieval. As shown in Table 2, it is natural that our v4 obtains high precision scores but relatively low recall scores due to the used rough cloud and shadow masks and its inherent edge-precise detection. We attribute the drop of categorical indicators to our embedded CRF refinement and its edge-precise segmentation property, as fully discussed previously: the refinement disposes of some plausible regions to obtain edge-precise refinement, which, on the other hand, eliminates some isolated regions; besides, the label references are not sufficiently precise at the pixel level, leading to an inferior recall performance of edge-precise segmentation. We further focus on the indicators of shadow detection. Interestingly, UNet × α obtains a higher R exceeding P as the adaptively weighted loss function leads the model to identify more but redundant shadow pixels at the same time. Edge-precise models, including Refined UNet, v2, v3, and our v4, consistently obtain higher P scores. We also attribute this to the property of our edge-precise segmentation. Finally, we conclude that our v4 is comparable to the involved counterparts in terms of the categorical evaluation. In addition, it should be noticed that the label references are not sufficiently precise at the pixel level, leading to valid relative comparisons rather than invalid absolute assessments; we shall further qualitatively evaluate our v4 from a more typical perspective.

Table 2. Average Precision, Recall, and F1 Scores on Our Landsat 8 OLI Test Set (Mean ± Standard Deviation, $^+$ represents that higher scores indicate better performance).

No.	**Models**	**Background (0)**			**Fill Value (1)**			**Shadow (2)**			**Cloud (3)**		
		P. $^+$ (%)	**R. $^+$ (%)**	**F1 $^+$ (%)**	**P. $^+$ (%)**	**R. $^+$ (%)**	**F1 $^+$ (%)**	**P. $^+$ (%)**	**R. $^+$ (%)**	**F1 $^+$ (%)**	**P. $^+$ (%)**	**R. $^+$ (%)**	**F1 $^+$ (%)**
1	**UNet** [1]	92.84 ± 5.81	81.83 ± 24.23	84.91 ± 20.54	100 ± 0	100 ± 0	100 ± 0	63.65 ± 38.27	5.35 ± 6.17	9.38 ± 10.27	80.39 ± 19.34	99.43 ± 0.87	87.45 ± 15.1
2	**PSPNet** [1]	65.49 ± 19.62	98.57 ± 2.18	77.06 ± 15.04	100 ± 0	95.97 ± 0.19	97.94 ± 0.1	46.81 ± 24.98	7.83 ± 5.95	12.74 ± 9.14	94.09 ± 17	48.22 ± 22.81	60.99 ± 22.56
3	**UNet** × α [2]	93.34 ± 4.88	81.52 ± 15.3	86.35 ± 11.04	100 ± 0	100 ± 0	100 ± 0	34.74 ± 14.77	54.31 ± 18.72	40.43 ± 14.74	87.28 ± 18.78	95.96 ± 3.63	90.12 ± 13.77
4	**Glo. R. UNet** [1]	89.89 ± 7.39	85.94 ± 17.66	86.86 ± 12.33	99.88 ± 0.07	100 ± 0	99.94 ± 0.04	35.43 ± 20.26	17.87 ± 12.07	21.21 ± 11.89	87.6 ± 19.15	95.87 ± 3.2	90.15 ± 14.13
5	**v2** ($r = 10$) [2]	91.99 ± 5.74	84.51 ± 16.24	87.29 ± 11.35	100 ± 0	100 ± 0	100 ± 0	40.64 ± 19.88	39 ± 13.77	36.79 ± 12.26	87.93 ± 18.83	95.83 ± 3.99	90.37 ± 13.89
6	**v2** ($r = 80$) [2]	89.57 ± 7.59	86.42 ± 18.25	86.75 ± 12.45	100 ± 0	100 ± 0	100 ± 0	32.22 ± 22.74	11.27 ± 8.16	13.85 ± 6.73	87.69 ± 19.01	95.06 ± 4.63	89.8 ± 13.99
7	**v3** ($r = 100$) [7]	90.48 ± 6.92	86.14 ± 17.9	87.16 ± 12.33	100 ± 0	100 ± 0	100 ± 0	37.79 ± 21.17	20.12 ± 10.38	23.46 ± 8.92	87.83 ± 18.98	95.69 ± 4.13	90.22 ± 13.99
8	**v4** ($\theta_\alpha = 60$) [1]	90.94 ± 6.41	85.61 ± 17.12	87.27 ± 11.93	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	38.85 ± 20.76	26.22 ± 11.37	28.72 ± 10.58	87.82 ± 18.89	95.88 ± 3.73	90.35 ± 13.95
9	**v4** ($\theta_\alpha = 80$)	90.75 ± 6.5	85.83 ± 17.26	87.28 ± 12.03	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	39.04 ± 21.08	24.32 ± 11.44	27.47 ± 10.79	87.8 ± 18.9	95.88 ± 3.7	90.34 ± 13.96
10	**v4** ($\theta_\alpha = 100$)	90.59 ± 6.59	86.01 ± 17.34	87.29 ± 12.07	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	39.16 ± 21.81	22.85 ± 11.53	26.33 ± 11.07	87.83 ± 18.9	95.82 ± 3.77	90.33 ± 13.95
11	**v4** ($\theta_\alpha = 120$)	90.48 ± 6.65	86.11 ± 17.47	87.28 ± 12.18	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	39.29 ± 22.06	21.75 ± 11.46	25.57 ± 11.46	87.82 ± 18.9	95.72 ± 3.87	90.27 ± 13.95
12	**v4** ($\theta_\alpha = 140$)	90.35 ± 6.7	86.24 ± 17.48	87.28 ± 12.2	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	39.34 ± 22.51	20.69 ± 11.46	24.61 ± 11.46	87.82 ± 18.89	95.72 ± 3.89	90.27 ± 13.93
13	**v4** ($\theta_\beta = 0.0625$) [2]	90.75 ± 6.5	85.83 ± 17.26	87.28 ± 12.03	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	39.04 ± 21.08	24.32 ± 11.44	27.47 ± 10.79	87.8 ± 18.9	95.88 ± 3.7	90.34 ± 13.96
14	**v4** ($\theta_\beta = 0.125$)	89.94 ± 7.02	86.42 ± 17.58	87.12 ± 12.1	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	38.89 ± 21.76	18.6 ± 9.84	22.41 ± 8.08	87.94 ± 18.95	95.05 ± 4.5	90.01 ± 13.99
15	**v4** ($\theta_\beta = 0.25$)	89.54 ± 7.6	86.86 ± 17.99	87.01 ± 12.23	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	34.92 ± 22.95	11.57 ± 7.5	14.71 ± 6.74	87.81 ± 18.99	95.09 ± 4.48	89.93 ± 13.99

[1] $\theta_\beta = 0.0625$. [2] $\theta_\alpha = 80$.

4.3. Visual Comparisons against Involved Methods

Next, we visually evaluate our Refined UNet v4 against other involved methods. It should be noted that the Refined UNet series [1,2,7] contribute to the edge refinement of cloud and shadow regions, which can be visually verified. Therefore, the visualizations are regarded as the principal evaluation rather than the quantitative indicators. Please note that we further restrict our attention to the segmentation performance around the edges of cloud and shadow regions, instead of merely considering coarsely locating the regions of interest; it is the distinguishing contribution of our current work. In order to illustrate the distinct contributions of our v4, the examples of full images and local patches with default $\theta_\alpha = 80$ and $\theta_\beta = 0.0625$ are exhibited in Figures 3–6. As shown in Figures 3 and 4, segmentation proposals are compared and evaluated in a full-resolution visual way. Naturally, all mentioned methods obtain visually similar segmentation proposals from the global perspective, which supports the conclusion that compared to other involved methods, our v4 can achieve comparable segmentation efficacy.

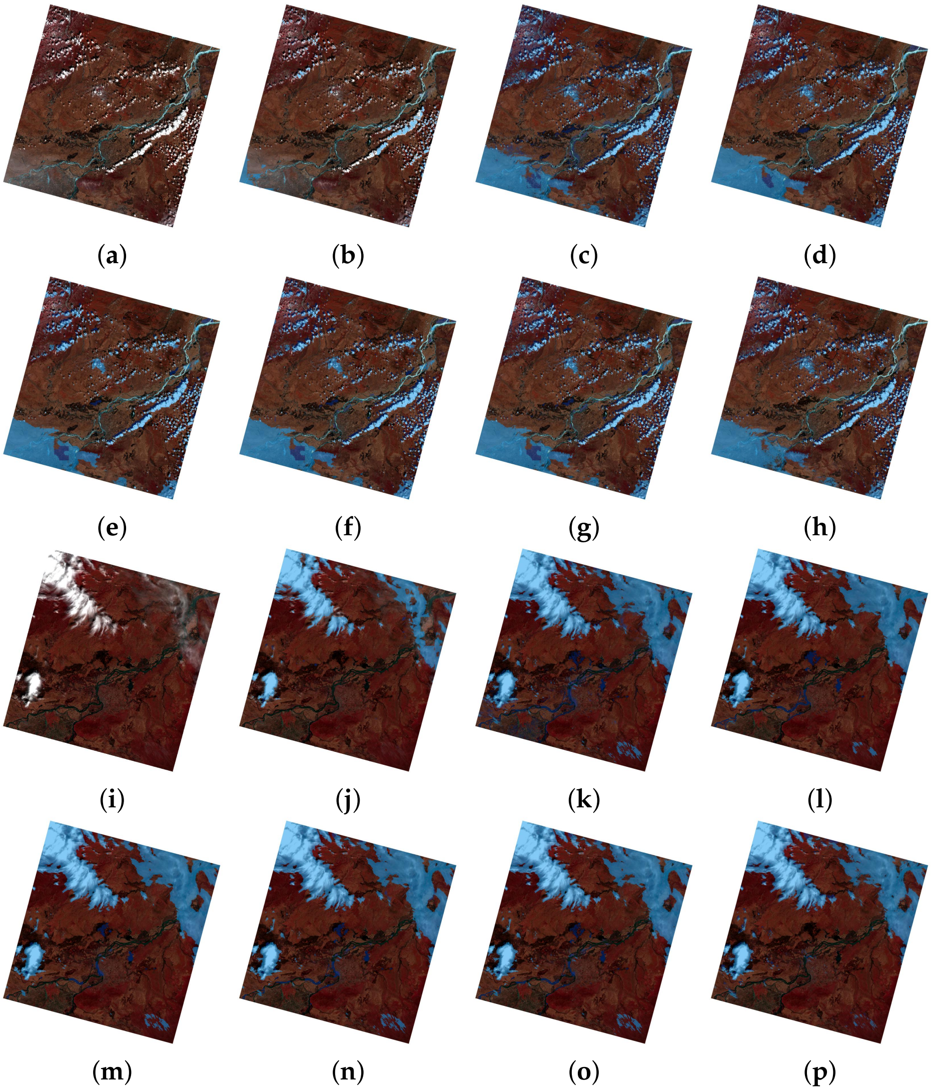
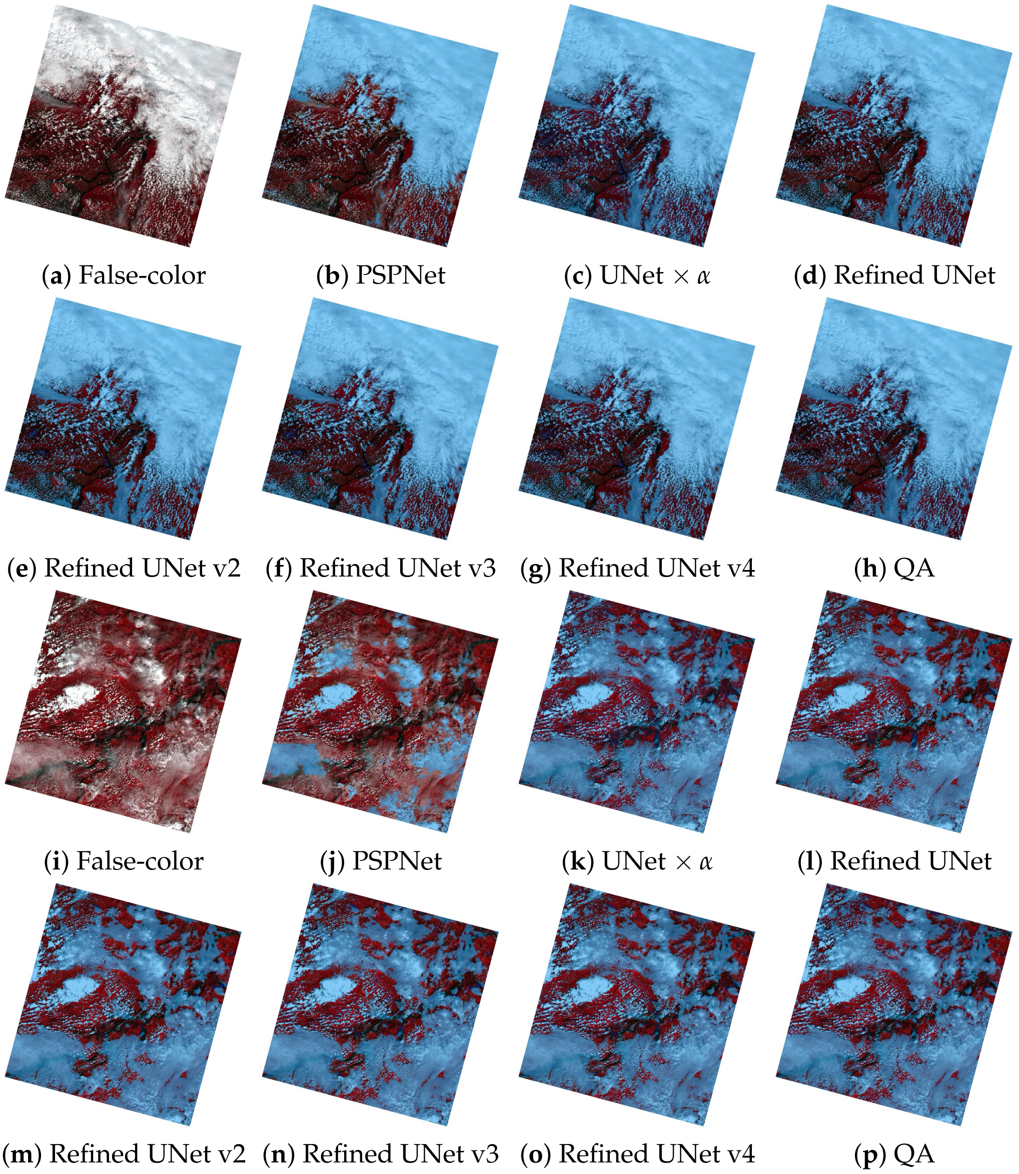


Figure 3. Typical visual examples of full images (1). (**a**,**i**) False-color (Bands 5 NIR, 4 Red, and 3 Green) images, (**b**,**j**) Reproduced PSPNet [1,31], (**c**,**k**) UNet × α [1], (**d**,**l**) Global Refined UNet [1], (**e**,**m**) Refined UNet v2 [2], (**f**,**n**) Refined UNet v3 [7], (**g**,**o**) Refined UNet v4, (**h**,**p**) QA reference. All comparative methods and our Refined UNet v4 have achieved similar coarse predictions for cloud and shadow regions, which demonstrates that our v4 is comparable to the reproduced instances of Refined UNets and other sophisticated models.



(**a**) False-color (**b**) PSPNet (**c**) UNet × α (**d**) Refined UNet

(**e**) Refined UNet v2 (**f**) Refined UNet v3 (**g**) Refined UNet v4 (**h**) QA

(**i**) False-color (**j**) PSPNet (**k**) UNet × α (**l**) Refined UNet

(**m**) Refined UNet v2 (**n**) Refined UNet v3 (**o**) Refined UNet v4 (**p**) QA

Figure 4. Typical visual examples of full images (2).

We further zoom in on some typical patches to observe the distinguishing contributions of our v4. As shown in Figures 5 and 6, notable superiority can be intuitively concluded. The reproduced PSPNet fetches cloud and shadow regions with extremely high confidence, which appears to be the innermost areas of interest. The UNet backbone conversely preserves ambiguous cloud and shadow regions, no matter the pixel belongs to cloud or snow. The previous Refined UNet series and our v4 make sufficient effort to refine the edges of cloud and shadow regions, but they perform in the different ways: Refined UNet v2 and v3 mainly contribute to the removal of isolated pixels and regions of interest, but they also aggressively eliminate some small regions that should be preserved; Refined UNet and our v4 are more sensitive to edges than v2 and v3 due to the regions of interest with tight edges, particularly for some cloud regions. On the other hand, v4 retrieves more shadow regions than v2 and v3, which evidences that v4 performs well on this hardly identified category. We attribute this to the superiority of the edge sensitivity of the Gaussian-form bilateral kernel over the locally linear kernel. In conclusion, our v4 is an end-to-end edge-precise solution satisfying the ultimate goal of neural segmentation, which is regarded as an encouraging improvement of the Refined UNet series.

Unfortunately, it is visually observed that comprehensive shadow detection is not sufficiently satisfied in our current work, and we can attribute this remaining issue to the approximation to the bilateral kernel discussed in the previous investigation [59]. We, therefore, will explore multiple approximations and suggest that all the crucial hyperparameters should be assigned in terms of the performance of the corresponding validation set when used in practice.

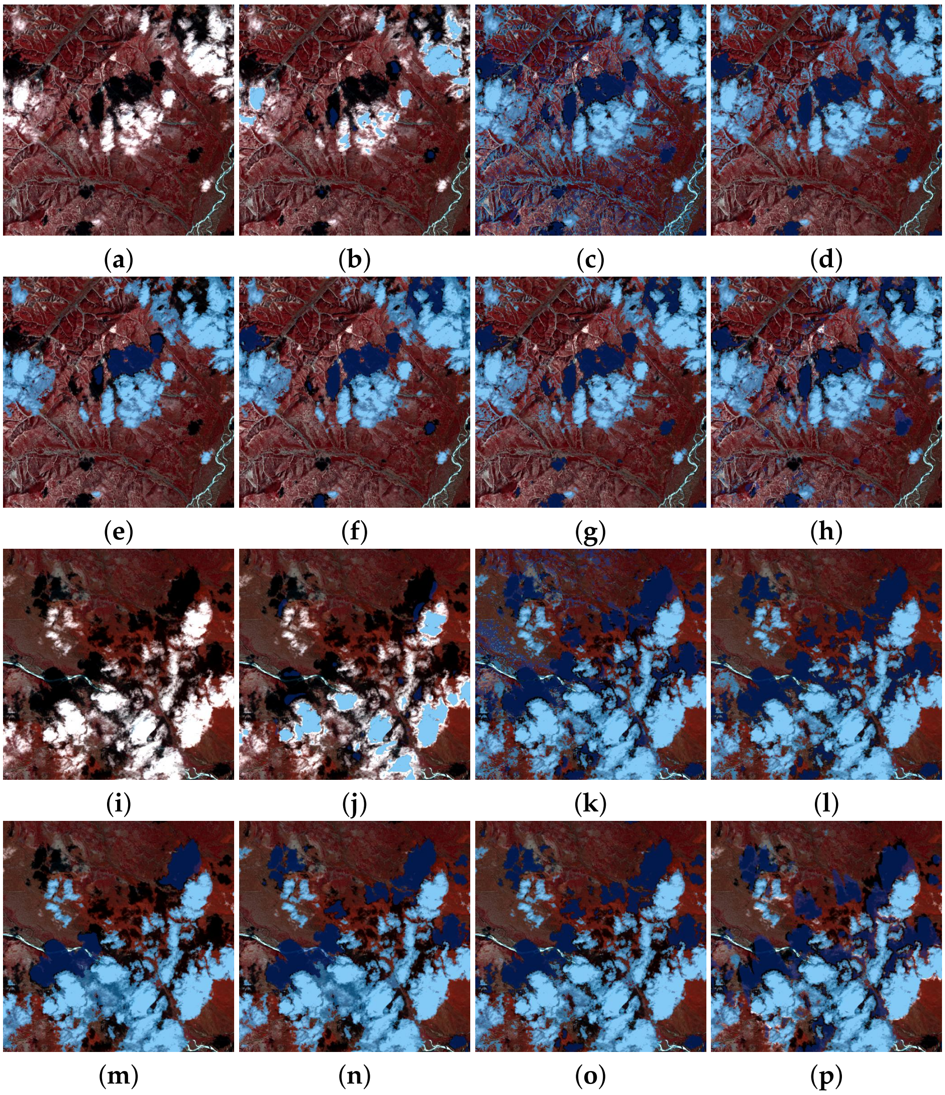
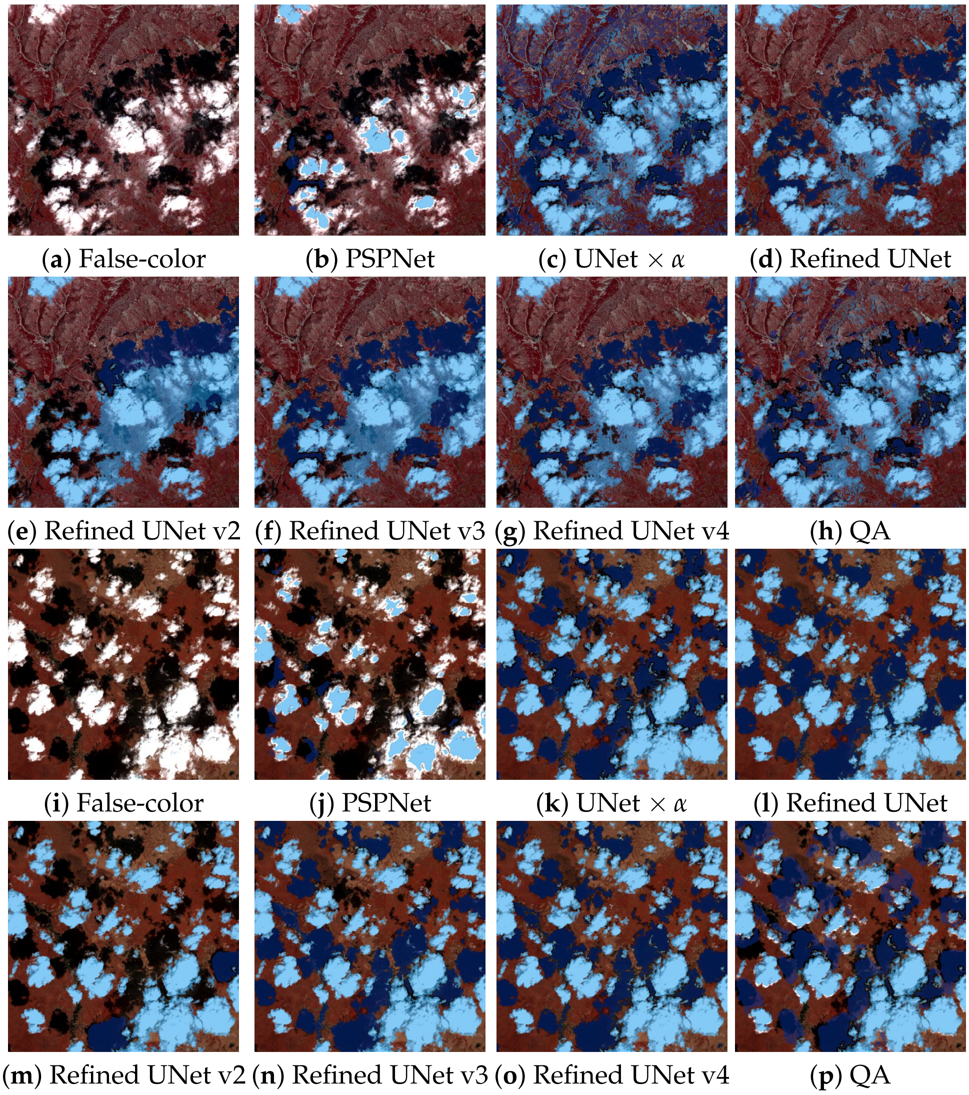


Figure 5. Typical visual examples of local patches (1). (**a**,**i**) False-color (Bands 5 NIR, 4 Red, and 3 Green) images, (**b**,**j**) Reproduced PSPNet [1,31], (**c**,**k**) UNet × α [1], (**d**,**l**) Global Refined UNet [1], (**e**,**m**) Refined UNet v2 [2], (**f**,**n**) Refined UNet v3 [7], (**g**,**o**) Refined UNet v4, (**h**,**p**) QA reference. Our Refined UNet v4 is able to yield tight and edge-precise segmentation proposals that are visually similar to that of Refined UNet [1], and it can also preserve ambiguous shadow regions, which qualitatively supports the distinguishing contributions.



(**a**) False-color (**b**) PSPNet (**c**) UNet × α (**d**) Refined UNet

(**e**) Refined UNet v2 (**f**) Refined UNet v3 (**g**) Refined UNet v4 (**h**) QA

(**i**) False-color (**j**) PSPNet (**k**) UNet × α (**l**) Refined UNet

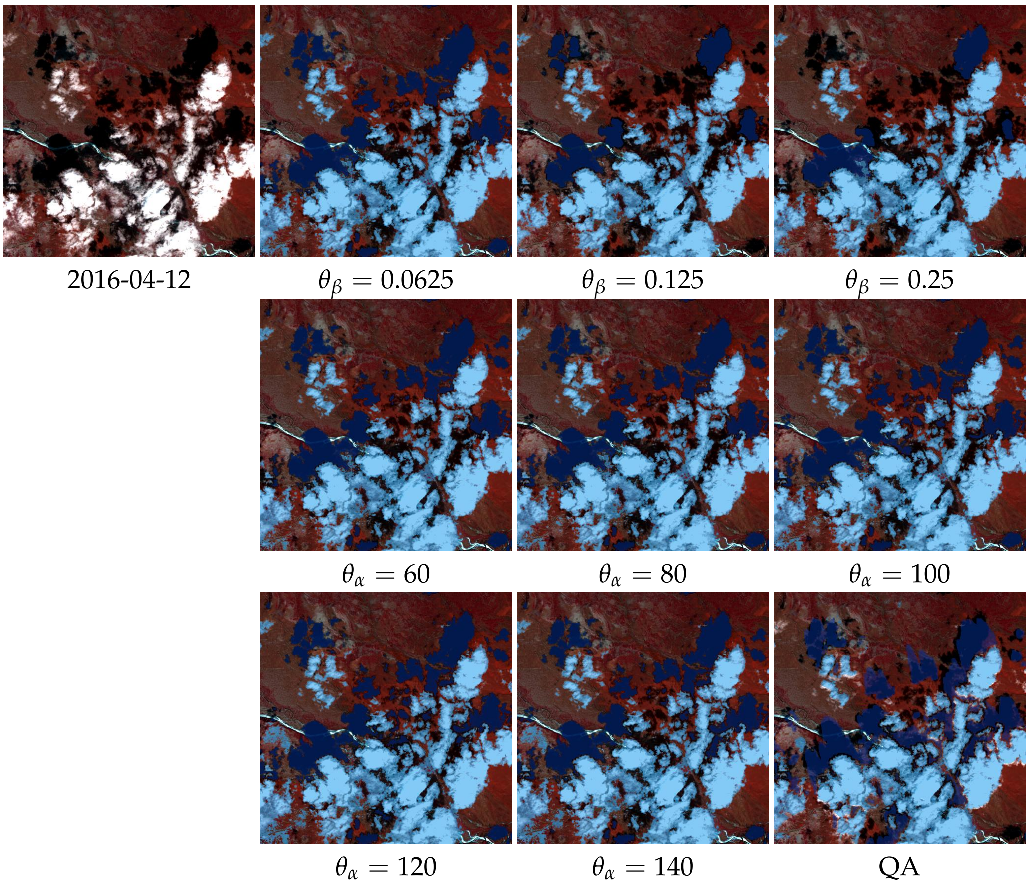
(**m**) Refined UNet v2 (**n**) Refined UNet v3 (**o**) Refined UNet v4 (**p**) QA

Figure 6. Typical visual examples of local patches (2).

4.4. Hyperparameter with Respect to θ_α and θ_β

We further evaluate the segmentation performance of our v4 with respect to its key hyperparameters θ_α and θ_β. As previously discussed in Dense CRF [47,48] and Refined UNet [1], a higher θ_α and a smaller θ_β yield more edge-precise segmentation proposals, and we will verify the aforementioned hypothesis by conducting experiments of varying parameters. Specifically, θ_α and θ_β vary from 60 to 140 by 20 and 0.0625 to 0.25 by 2, respectively. We first investigate the statistical variations of evaluation metrics, as shown in Tables 1 and 2. We first discuss the segmentation performance with respect to θ_α. According to lines 8 to 12 in Table 1, there are no significant differences observed for overall accuracy and Kappa, whereas mean IoUs drop slightly. As reported in Table 2, the precision scores of interest keep stable but the recall scores of shadow decrease significantly. We attribute this drop to the inherent property of shadow refinement: CRF with a higher θ_α is able to refine edges of shadow regions effectively but it also removes more ambiguous isolated areas; intuitively, it performs more conservatively. Second, we turn to the discussion with respect to θ_β. According to lines 13 to 15 in both Tables 1 and 2, a smaller θ_β obtains higher overall and categorical indicators, leading to a significantly better segmentation result. We attribute this performance to the nature of our bilateral step that smaller θ_β is more sensitive to edges, which tends to preserve possible edges and potential regions.

In addition, we verify this hypothesis from the locally visual perspective. We observe the visual variation in one particular patch, shown in Figure 7. As illustrated, CRF with a smaller θ_β delineates edges of regions of interest more precisely and retrieves more shadow regions, which confirms the property that a smaller θ_β is able to preserve possible edges and potential regions. Whereas a larger θ_β is not edge-sensitive enough; it alternatively removes more isolated pixels and regions such that it visually smooths the proposals. However, θ_α has a limited effect on the edge refinement, leading to similar segmentation proposals. We can confirm the hypothesis that a smaller θ_β gives rise to better segmentation results.



2016-04-12 $\theta_\beta = 0.0625$ $\theta_\beta = 0.125$ $\theta_\beta = 0.25$

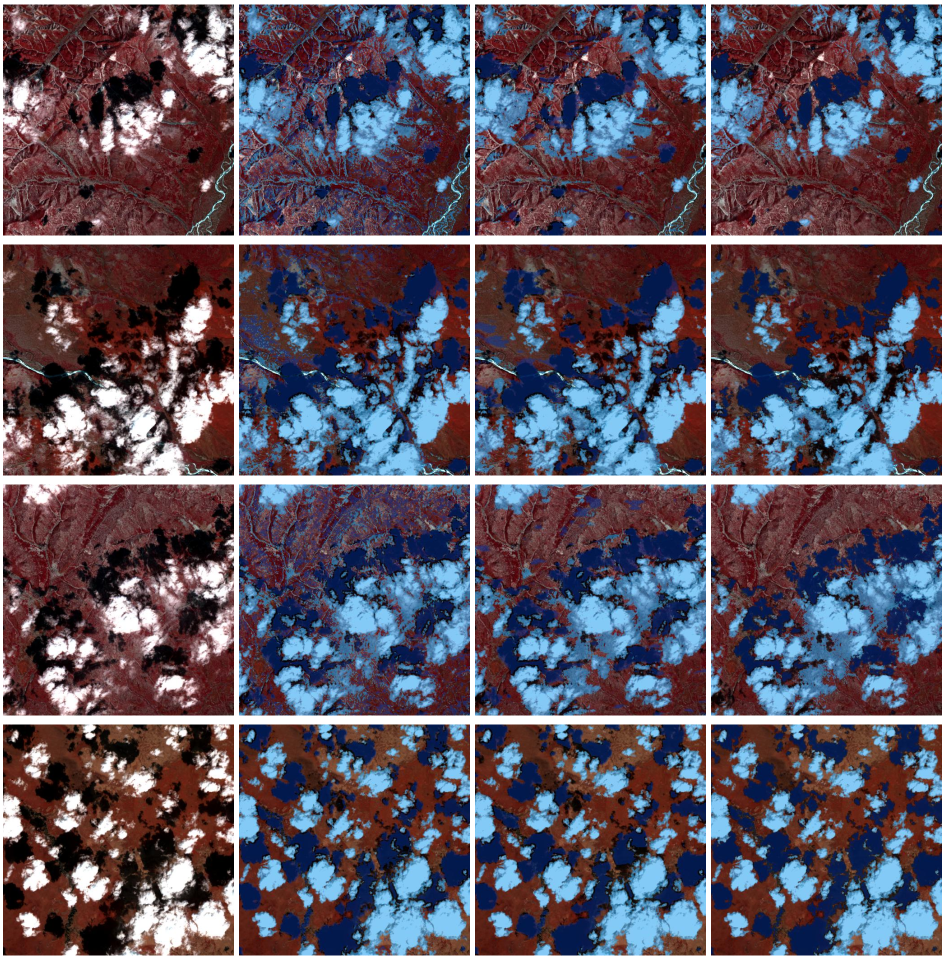
$\theta_\alpha = 60$ $\theta_\alpha = 80$ $\theta_\alpha = 100$

$\theta_\alpha = 120$ $\theta_\alpha = 140$ QA

Figure 7. Typical visual examples in the hyperparameter test. We experiment with two key parameters θ_α and θ_β and visually evaluate their effects. CRF with a smaller θ_β is more sensitive to edges, leading to more edge-precise proposals and tolerant preservation of potential shadow regions. On the other hand, θ_α has a visually insignificant effect on the edge refinement.

4.5. *Ablation Study with Respect to Our Gaussian-Form Bilateral Approximation*

We verify the effect of our CRF layer and the bilateral approximation to its Gaussian-form bilateral kernel with the ablation study in this subsection. As we discussed previously, the UNet backbone achieves coarse-grained location of regions of interest and the embedded CRF layer significantly delineates the corresponding edges; in particular, the bilateral kernel plays an important role in edge refinement. We, therefore, present the illustrations to verify the effect of our CRF layer and the approximation to its Gaussian-form bilateral kernel, shown in Figure 8. As shown in Figure 8, the UNet backbone is effectively able to coarsely locate regions of interest but fails in fine-grained refinement and preserves redundant isolated pixels and regions. UNet-CRF without bilateral message-passing steps performs similarly but removes in part redundant detection noises. Instead, our full v4 is able to achieve the edge-precise detection of regions of interest and to denoise the proposal as well, in terms of significant visual superiority. The ablation study confirms that our CRF layer and the approximation to Gaussian-form bilateral kernel have dramatically visual contributions to the edge-precise cloud and shadow detection.



(**a**) False-color (**b**) UNet backbone (**c**) V4 w/o bilateral (**d**) V4

Figure 8. Illustrations of our ablation study. (**a**) False-color (Bands 5 NIR, 4 Red, and 3 Green) images, (**b**) The UNet backbone (UNet × α) [1], (**c**) Refined UNet v4 without the bilateral kernel, (**d**) Refined UNet v4. The UNet backbone can coarsely locate cloud and shadow regions, and v4 without the bilateral kernel helps remove redundant isolated detection pixels and regions. Our full v4, instead, effectively achieves edge-precise detection and denoises proposals.

4.6. Computational Efficiency of Refined UNet v4

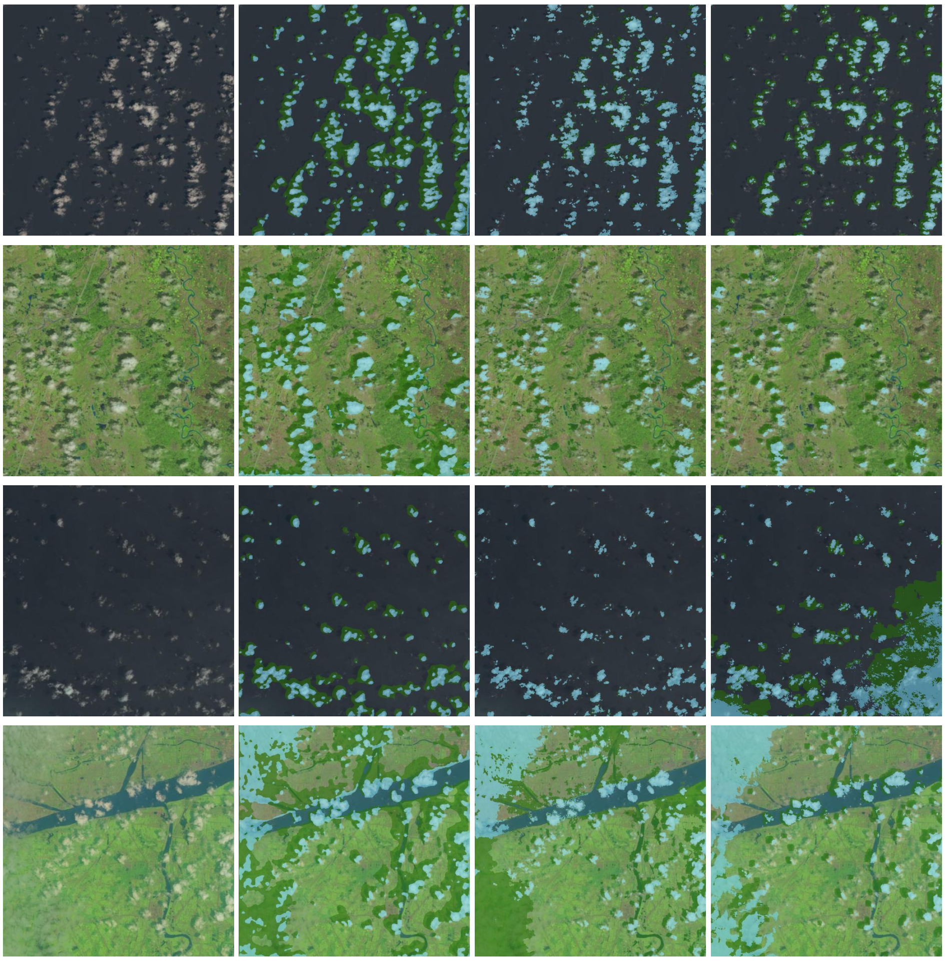
We evaluate the time consumption of our v4 in this subsection. We first compare the consumption of our v4 with Refined UNet, v2, and efficient v3. As indicated in Table 1, our v4 is relatively efficient in comparison with Refined UNet but is unfortunately left behind v3. We attribute this to our naïve implementation of the bilateral message-passing step, which should be improved in future work. Considering that our v4 visually outperforms v2 and v3 with guided Gaussian filter-based bilateral steps, we believe that the edge-precision gain of v4 exceeds its computational cost.

We further evaluate the consumption with respect to the key hyperparameters θ_α and θ_β. As indicated in Table 1, θ_α has a significant effect on computational cost, concluded by the drop in the time consumptions. It is surely because a higher θ_α leads to relatively sparse sampling of splatting and a lower-scale convolution of blurring, which produces the time-saving result. Alternatively, it is observed from Table 1 that θ_β slightly affects the computational consumption; its nature is likewise attributed to the scales of sampling of splatting and of the convolution of blurring. Considering that a lower θ_β induces more edge-precise segmentation proposals, we also believe that the edge-precision gain with lower θ_β matters a lot more compared to the possible slight increase of the computational cost.

4.7. Generalization to RICE Dataset

We generalize our Refined UNet v4 to the cloud and shadow detection on the RICE dataset [65] that is available at https://github.com/BUPTLdy/RICE_DATASET (accessed on 21 December 2021). We train the UNet backbone from scratch and infer the cloud and shadow regions with the full v4. The experimental setups are listed as follows: the input resolution is 512 × 512. The loss function is the adaptively weighted categorical cross-entropy function, introduced in [1]. The initial learning rate, the regularization factor, the decay step, and the batch size in the training phase are 0.0001, 0.0001, 100, and 1, respectively. The optimizer is ADAM [66]. We observe that the validation loss and accuracy converge after 50 epochs so we stop training early and secure the parameters of the UNet. In the inference phase, we empirically assign θ_α, θ_β, and θ_γ to 80, 0.0625, and 3, respectively. Please also refer to our GitHub repository for more training and inference details.

Figure 9 illustrates the natural images, detection proposals from UNet, refined proposals from v4, and detection references, respectively. We can find that our v4 is sufficiently sensitive to the edges and yields proposals with tight edges. It should be also noted that we refer to the used labels as references rather than ground truths because these labels are also not sufficiently precise at the pixel level. However, our v4 also yields edge-precise proposals, partially contributing to the weakly supervised cloud and shadow detections.



(**a**) Natural (**b**) UNet backbone (**c**) RFN. UNet v4 (**d**) Reference

Figure 9. Detection generalization to RICE dataset. (**a**) Natural images, (**b**) UNet backbone, (**c**) Refined UNet v4, (**d**) Reference. Visually, our v4 yields edge-precise proposals for cloud and shadow regions even though our training labels are not sufficiently precise at the pixel level. Comprehensive retrievals of shadow regions with ambiguous edges and cirrus regions, however, remain challenging.

Unfortunately, we have to admit that our v4 can rarely identify all the regions of interest in a comprehensive way: some shadow regions with ambiguous edges remain missing in our detection proposals, and some cirrus regions are hardly identified by both our model and even artificial differentiation. This is possibly attributed to the inherent segmentation property of our v4, which will be considered in our future improvement.

5. Conclusions

In this paper, we present Refined UNet v4, an end-to-end segmentation network for edge-precise detection of cloud and shadow regions. Our v4 inherits the pretrained UNet backbone to coarsely locate the cloud and shadow regions and subsequently concatenates a dedicated CRF layer to refine the edges of the regions of interest; the aforementioned steps form an end-to-end segmentation solution, which enables cloud and shadow segmentation in a one-stage way. In particular, the bilateral grid-based high dimensional filter is adopted to facilitate the relatively efficient bilateral message-passing step of the embedded CRF layer, leading to a balanced trade-off between edge-precision and computational consumption. The extensive experiments are conducted on the test set drawn from the Landsat 8 OLI remote sensing dataset, comprehensively evaluating our v4 from both quantitative and visual perspectives. The quantitative evaluations indicate that our v4 is comparable to its counterparts of the Refined UNet series, while the visual evaluations highlight its merits on the edge-precision of cloud and shadow detection: tight edges and relatively sufficient shadow retrieval. In addition, the hyperparameter tests demonstrate that the range parameter (θ_β) has a significant visual impact on the sensitivity of edge sensing: a smaller θ_β preserves more regions of interest and delineates more precisely, whereas a larger θ_β is not edge-sensitive enough; it alternatively removes more isolated pixels and regions such that it visually smooths the proposals. The ablation study confirms that our bilateral approximation to the bilateral message-passing step plays a crucial role in obtaining edge-precise proposals. We test and compare the time consumption of the involved methods, indicating that our v4 is relatively computationally efficient compared with the global Refined UNet; we attribute this nature to the straightforward GPU support of its TensorFlow implementation. We finally generalize our model to the RICE dataset and conclude that our model has a relatively satisfactory generalization and reproductivity. On the other hand, we are also concerned about the learnability of UNet-CRF architecture, which will be discussed in our future work.

Author Contributions: Conceptualization, L.J. and P.T.; Funding acquisition, L.H. and P.T.; Methodology, L.J.; Supervision, L.H., C.H., P.T. and Z.Z.; Writing—original draft, L.J.; Writing—review & editing, L.H., C.H. and Z.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China grant number 41971396 and Strategic Priority Research Program of the Chinese Academy of Sciences grant number XDA19080301.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank for the open-source implementations of the fast bilateral filter (http://people.csail.mit.edu/sparis/bf/ (accessed on 22 September 2021)) as they inspired the implementation of Refined UNet v4. A pure TensorFlow implementation of Refined UNet v4 is publicly available at https://github.com/92xianshen/refined-unet-v4 (accessed on 22 December 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jiao, L.; Huo, L.; Hu, C.; Tang, P. Refined UNet: UNet-Based Refinement Network for Cloud and Shadow Precise Segmentation. *Remote Sens.* **2020**, *12*, 2001. [CrossRef]
2. Jiao, L.; Huo, L.; Hu, C.; Tang, P. Refined UNet V2: End-to-End Patch-Wise Network for Noise-Free Cloud and Shadow Segmentation. *Remote Sens.* **2020**, *12*, 3530. [CrossRef]

3. Roy, D.P.; Wulder, M.A.; Loveland, T.R.; Woodcock, C.E.; Zhu, Z. Landsat-8: Science and product vision for terrestrial global change research. *Remote Sens. Environ.* **2014**, *145*, 154–172. [CrossRef]
4. Wulder, M.A.; White, J.C.; Loveland, T.R.; Woodcock, C.E.; Roy, D.P. The global Landsat archive: Status, consolidation, and direction. *Remote Sens. Environ.* **2016**, *185*, 271–283. [CrossRef]
5. Vermote, E.F.; Justice, C.O.; Claverie, M.; Franch, B. Preliminary analysis of the performance of the Landsat 8/OLI land surface reflectance product. *Remote Sens. Environ.* **2016**, *185*, 46–56. [CrossRef] [PubMed]
6. Chai, D.; Newsam, S.; Zhang, H.K.; Qiu, Y.; Huang, J. Cloud and cloud shadow detection in Landsat imagery based on deep convolutional neural networks. *Remote Sens. Environ.* **2019**, *225*, 307–316. [CrossRef]
7. Jiao, L.; Huo, L.; Hu, C.; Tang, P. Refined UNet v3: Efficient end-to-end patch-wise network for cloud and shadow segmentation with multi-channel spectral features. *Neural Netw.* **2021**, *143*, 767–782. [CrossRef]
8. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *39*, 640–651.
9. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.A. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
10. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
11. Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
12. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
13. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for MobileNetV3. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019.
14. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity Mappings in Deep Residual Networks. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 630–645.
16. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269.
17. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning, ICML'19, Long Beach, CA, USA, 9–15 June 2019; pp. 6105–6114.
18. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*; Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 234–241.
19. Chen, L.C.; Yang, Y.; Wang, J.; Xu, W.; Yuille, A.L. Attention to Scale: Scale-Aware Semantic Image Segmentation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
20. Farabet, C.; Couprie, C.; Najman, L.; Lecun, Y. Learning Hierarchical Features for Scene Labeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1915–1929. [CrossRef]
21. Mostajabi, M.; Yadollahpour, P.; Shakhnarovich, G. Feedforward semantic segmentation with zoom-out features. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
22. Zhang, H.; Dana, K.; Shi, J.; Zhang, Z.; Wang, X.; Tyagi, A.; Agrawal, A. Context Encoding for Semantic Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
23. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *arXiv* **2014**, arXiv:1412.7062.
24. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef]
25. Chen, L.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.
26. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
27. Li, H.; Xiong, P.; An, J.; Wang, L. Pyramid Attention Network for Semantic Segmentation. *arXiv* **2018**, arXiv:1805.10180.
28. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions. *arXiv* **2016**, arXiv:1511.07122.
29. Lin, G.; Milan, A.; Shen, C.; Reid, I. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
30. Lin, G.; Shen, C.; Hengel, A.V.D.; Reid, I. Efficient piecewise training of deep structured models for semantic segmentation. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
31. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.

32. Liu, W.; Rabinovich, A.; Berg, A. ParseNet: Looking Wider to See Better. *arXiv* **2015**, arXiv:1506.04579.
33. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
34. Kendall, A.; Badrinarayanan, V.; Cipolla, R. Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. In Proceedings of the British Machine Vision Conference, London, UK, 4–7 September 2017.
35. Wu, H.; Zhang, J.; Huang, K.; Liang, K.; Yu, Y. FastFCN: Rethinking Dilated Convolution in the Backbone for Semantic Segmentation. *arXiv* **2019**, arXiv:1903.11816.
36. Wang, P.; Chen, P.; Yuan, Y.; Liu, D.; Huang, Z.; Hou, X.; Cottrell, G. Understanding Convolution for Semantic Segmentation. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018.
37. Sun, L.; Wang, J.; Yang, K.; Wu, K.; Zhou, X.; Wang, K.; Bai, J. Aerial-PASS: Panoramic Annular Scene Segmentation in Drone Videos. In Proceedings of the 2021 European Conference on Mobile Robots (ECMR), Bonn, Germany, 31 August–3 September 2021; pp. 1–6. [CrossRef]
38. Li, X.; He, H.; Li, X.; Li, D.; Cheng, G.; Shi, J.; Weng, L.; Tong, Y.; Lin, Z. PointFlow: Flowing Semantics Through Points for Aerial Image Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 4217–4226.
39. Strudel, R.; Pinel, R.G.; Laptev, I.; Schmid, C. Segmenter: Transformer for Semantic Segmentation. *arXiv* **2021**, arXiv:2105.05633.
40. Wang, Y.; Xu, Z.; Wang, X.; Shen, C.; Cheng, B.; Shen, H.; Xia, H. End-to-End Video Instance Segmentation with Transformers. *arXiv* **2020**, arXiv:2011.14503,
41. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.H.S.; et al. Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers. *arXiv* **2020**, arXiv:2012.15840.
42. Petit, O.; Thome, N.; Rambour, C.; Soler, L. U-Net Transformer: Self and Cross Attention for Medical Image Segmentation. *arXiv* **2021**, arXiv:2103.06104.
43. Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J.M.; Luo, P. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. *arXiv* **2021**, arXiv:2105.15203.
44. Chen, J.; Lu, Y.; Yu, Q.; Luo, X.; Adeli, E.; Wang, Y.; Lu, L.; Yuille, A.L.; Zhou, Y. TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation. *arXiv* **2021**, arXiv:2102.04306.
45. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *arXiv* **2021**, arXiv:2103.14030.
46. Zhang, H.; Patel, V.M. Densely Connected Pyramid Dehazing Network. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
47. Krähenbühl, P.; Koltun, V. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In *Advances in Neural Information Processing Systems 24*; Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q., Eds.; 2011; pp. 109–117.
48. Krähenbühl, P.; Koltun, V. Parameter Learning and Convergent Inference for Dense Random Fields. In Proceedings of the 30th International Conference on Machine Learning, ICML'13, Atlanta, GA, USA, 17–19 June 2013; pp. 513–521.
49. Zheng, S.; Jayasumana, S.; Romeraparedes, B.; Vineet, V.; Su, Z.; Du, D.; Huang, C.; Torr, P.H.S. Conditional Random Fields as Recurrent Neural Networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 1529–1537.
50. Liu, Z.; Li, X.; Luo, P.; Loy, C.C.; Tang, X. Semantic Image Segmentation via Deep Parsing Network. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 11–18 December 2015.
51. Richard, X.H.; Miguel, S.Z.; An, C.P.N. Multiscale conditional random fields for image labeling. *Proc. IEEE Comput. Vis. Patern Recognit.* **2004**, *2*, II–695–II–702.
52. He, K.; Sun, J.; Tang, X. Guided Image Filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1397–1409. [CrossRef]
53. Wu, H.; Zheng, S.; Zhang, J.; Huang, K. Fast End-to-End Trainable Guided Filter. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 1838–1847.
54. Porikli, F. Constant time $O(1)$ bilateral filtering. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008.
55. Chaudhury, K.N.; Sage, D.; Unser, M. Fast $O(1)$ Bilateral Filtering Using Trigonometric Range Kernels. *IEEE Trans. Image Process.* **2011**, *20*, 3376–3382. [CrossRef]
56. Weiss, B. Fast median and bilateral filtering. *Acm Trans. Graph.* **2006**, *25*, 519–526. [CrossRef]
57. Yang, Q.; Tan, K.H.; Ahuja, N. Real-time $O(1)$ bilateral filtering. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009.
58. Durand, F.; Dorsey, J. Fast Bilateral Filtering for the Display of High-Dynamic-Range Images. *Acm Trans Graph.* **2002**, *21*, 257–266. [CrossRef]
59. Paris, S.; Durand, F. A Fast Approximation of the Bilateral Filter Using a Signal Processing Approach. *Int. J. Comput. Vis.* **2009**, *81*, 24–52. [CrossRef]
60. Adams, A.; Baek, J.; Davis, M.A. Fast High-Dimensional Filtering Using the Permutohedral Lattice. *Comput. Graph. Forum* **2010**, *29*, 753–762. [CrossRef]

61. Adams, A.; Gelfand, N.; Dolson, J.; Levoy, M. Gaussian KD-trees for fast high-dimensional filtering. *ACM Trans. Graph.* **2009**, *28*, 1–12. [CrossRef]
62. Chen, J.; Paris, S.; Durand, F. Real-time edge-aware image processing with the bilateral grid. *ACM Trans. Graph.* **2007**, *26*, 103. [CrossRef]
63. Chen, J.; Adams, A.; Wadhwa, N.; Hasinoff, S.W. Bilateral guided upsampling. *Acm Trans. Graph.* **2016**, *35*, 203. [CrossRef]
64. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. 2015. Available online: tensorflow.org (accessed on 27 October 2020).
65. Lin, D.; Xu, G.; Wang, X.; Wang, Y.; Sun, X.; Fu, K. A Remote Sensing Image Dataset for Cloud Removal. *arXiv* **2019**, arXiv:1901.00600.
66. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.