sensors

MDPI

Article

MSDU-Net: A Multi-Scale Dilated U-Net for Blur Detection

Xiao Xiao [1,*], Fan Yang [1] and Amir Sadovnik [2]

[1] School of Telecommunications Engineering, Xidian University, Xi'an 710071, China; fyang_jfsl@stu.xidian.edu.cn
[2] Department of Electrical Engineering & Computer Science, The University of Tennessee, Knoxville, TN 37996, USA; asadovnik@utk.edu
* Correspondence: xiaoxiao@xidian.edu.cn

Abstract: A blur detection problem which aims to separate the blurred and clear regions of an image is widely used in many important computer vision tasks such object detection, semantic segmentation, and face recognition, attracting increasing attention from researchers and industry in recent years. To improve the quality of the image separation, many researchers have spent enormous efforts on extracting features from various scales of images. However, the matter of how to extract blur features and fuse these features synchronously is still a big challenge. In this paper, we regard blur detection as an image segmentation problem. Inspired by the success of the U-net architecture for image segmentation, we propose a multi-scale dilated convolutional neural network called MSDU-net. In this model, we design a group of multi-scale feature extractors with dilated convolutions to extract textual information at different scales at the same time. The U-shape architecture of the MSDU-net can fuse the different-scale texture features and generated semantic features to support the image segmentation task. We conduct extensive experiments on two classic public benchmark datasets and show that the MSDU-net outperforms other state-of-the-art blur detection approaches.

Keywords: blur detection; image segmentation; U-shaped network

check for updates

Citation: Xiao, X.; Yang, F.; Sadovnik, A. MSDU-Net: A Multi-Scale Dilated U-Net for Blur Detection. *Sensors* **2021**, *21*, 1873. https://doi.org/10.3390/s21051873

Academic Editor: Sonya A. Coleman

Received: 18 January 2021
Accepted: 2 March 2021
Published: 8 March 2021

1. Introduction

Image blurring is one of the most common types of degradation caused by the relative motion between the sensor and the scene during image capturing. Object motion, camera shake, or objects being out of focus will cause the image to be blurred and reduce the visual quality of the image. This procedure can be regarded as the convolution of a clear image and a blur kernel, which is shown as:

$$\mathbf{B} = \mathbf{I} \otimes \mathbf{K} + \mathbf{N} \tag{1}$$

where **B** is the blurred image, **I** is the clear image, **K** is the blur kernel, ⊗ is the convolution, and **N** is the noise. Since the blur kernel is usually unknown and varies greatly in size, weight, shape, and position, the estimation of blur kernel is an ill-posed inverse problem. The first important step for blur estimation is to detect the blurred regions in an image and separate them from clear regions. Image blurring can be categorized into two main types: defocus blur caused by defocusing and motion blur caused by camera or object motion. Blur detection plays a significant role in many potential applications, such as salient object detection [1,2], defocus magnification [3,4], image quality assessment [5,6], image deblurring [7], image refocusing [8,9], and blur reconstruction [10,11].

In the past few decades, a series of blur detection methods based on hand-crafted features have been proposed. These methods exploit various hand-crafted features related to the image gradient [12–15] and frequency [3,7,16,17]. They tend to measure the amount of feature information contained in different image regions to detect blurriness, as the blurred regions usually contain fewer details than the sharp ones. However, these hand-crafted features are usually not good at differentiating sharp regions from a complex background and cannot understand semantics to extract sharp regions from a similar background.

Recently, deep convolutional neural networks (DCNNs) have made vital contributions to various computer vision tasks, such as image classification [18,19], object detection [20,21] and tracking [22,23], image segmentation [24,25], image denoising [26,27], image interpolation [28], and super resolution [29,30]. Several DCNN-based methods have been proposed thus far to address blur detection [31–36]. Some methods [35,36] use patch-level DCNNs to learn blur features in every patch, and others [31–34] use a fully convolutional network trained at different scales to learn blur features from multiple scales. They use many different types of various extractors to capture the essential feature information to detect blur information. In this paper, from a different perspective we consider blur detection as an image segmentation problem. Hence, we can learn some successful methods and tricks from the image segmentation area. Inspired by some classical image segmentation approaches such as U-net [25], we propose a U-shape multi-scale dilated network called MSDU-net to detect blurred regions. In this model, the U-shape architecture uses skip connections to combine the shallow features and deep features smoothly. Moreover, this model can fully utilize of the texture and semantic features of an image. In this work, we find that texture information can be used to describe the degree of blur, and semantic information plays a vital role in measuring the blurriness of each region in an image. Therefore, we propose a group of multi-scale feature extractors to capture different-scale texture features in a synchronous manner. Additionally, We apply dilated convolution with various dilation rates and strides to capture the texture information with different receptive fields [37,38]. In particular, we use low-dilation convolution and small-stride convolution to capture the texture information on a small scale and use a high dilation rate convolution and a large stride convolution to capture texture information on a large scale.

To sum up, our main contributions are as follows:

- We proposed a group of extractors with dilated convolution to capture multi-scale texture information on purpose rather than using a fully convolutional network multiple times on the different scales of the image.
- We designed a new model with our extractors based on U-net, which can fuse the multi-scale texture and semantic features simultaneously to improve the accuracy.
- Most methods only can detect the defocus blur or the motion blur, but our method addressed the blur detection, ignoring the specific cause of the blur and thus could detect both defocus blur and motion blur. Compared with the state-of-the-art blur detection methods, the proposed model obtained $F_{\sqrt{0.3}}$-measure scores of more than 95% in all the three datasets.

The rest of the paper is organized as follows: In Section 2, we introduce the traditional methods and deep learning methods and also some successful methods for image segmentation. In Section 3, we propose our model and describe the details of the neural network. In Section 4, we use our model with public blur detection datasets and compare our experimental results with those of other state-of-the-art methods. In Section 5, we conclude all the work of the paper.

2. Related Work

In this section, we will introduce the related work in the area of blur detection. We will show two main streams: (1) traditional views of blur detection and (2) regarding the blur detection problem as an image segmentation problem.

Previous methods of blur detection can be divided into two categories: methods based on traditional hand-crafted features and methods based on deep learning neural networks. In the first category, various hand-crafted features exploit gradient and frequency and can describe the information of regions. For example, Su et al. [39] used the gradient distribution pattern of the alpha channel and a metric based on singular value distributions together to detect the blurred region. In 2014, Shi et al. [7] made use of a series of gradient, Fourier domain, and data-driven local filter features to enhance the discriminative power for blur detection. In 2015, to enable feature extractors to distinguish noticeable blur reliably from unblurred structures, Shi et al. [16] improved feature extractors via sparse representation and image decomposition. Yi et al. [12] used a designed metric based on local binary patterns to detect the defocus regions. Tang et al. [17] designed a log-averaged spectrum residual metric to obtain a coarse blur map and iterate to a fine result based on the regional correlation. Golestaneh et al. [13] used a discrete cosine transform based on a high-frequency multi-scale fusion and sorted the transform of gradient magnitudes to detect a blurred region. In summary, traditional approaches aim to improve the accuracy with more meaningful and representative hand-crafted features, which are more interpretable. However, designing such features is difficult and the performance various among different datasets.

Because of the outstanding performance in high-level feature extraction and parameter learning, deep convolutional neural networks have reached a new state-of-the-art level in blur detection. Firstly, Park et al. [36] and Huang et al. [35] both used patch-level DCNNs in their methods to caputre local features more robustly to help detect blurred regions. Although patch-level DCNN methods use DCNNs, they do not make full use of the advantages of DCNNs. In 2018, Zhao et al. proposed a multi-stream bottom-top-bottom fully convolutional network [40], and Ma et al. also proposed an end-to-end fully convolution network [33]. Both the two methods are based on fully convolutional networks, and they novelly use high-level semantic information to help with blur detection. In order to increase the efficiency of the network, Tang et al. proposed a new blur detection deep neural network [34] by recurrently fusing and refining multi-scale features. Zhao et al. designed a cross-ensemble network [32] with two groups of defocus blur detectors, which were alternately optimized with cross-negative and self-negative correlation losses to enhance the diversity of features. With the application of DCNNs in computer vision, more solutions have been proposed for blur detection. Making the network deeper or wider to catch more useful features has been proven to be possible, but this kind of method is so dull that it incurs unnecessary resource consumption.

Except for the traditional view of the blur detection problem, blur detection problems can also be regarded as image segmentation problems. As we know, fully convolutional networks (FCNs) [41], which train end-to-end and pixel-to-pixel on semantic segmentation, exceed the previous best results without further machinery. Some classical architectures, such as DeepLab models [42–44] and U-net [25], have good performance in image segmentation. In DeepLab models [42–44], dilated convolution has been used to efficiently obtain feature maps with a larger receptive field.

The U-shape network was first proposed in [25] to address biomedical image segmentation, and it only had a few training samples. To make the best use of the limited samples, U-net [25] combines skip layers and learned deconvolution to fuse the different-level features of one image for a more precise result. Because of its outstanding performance in biomedical datasets with simple semantic information and a few fixed features, there are many further studies based on it, such as VNet [45], which is a U-shaped network that uses three-dimensional convolutions; UNet++ [46], which is a U-shaped network with more dense skip connections; Attention U-net [47], which combines U-shaped networks with an attention mechanism; ResUNet [48], which implements a U-shaped network with residual convolution blocks; TernausNet [49], which uses the pre-trained encoder to improve a U-shape network; MDU-Net [50], which densely connects the many scales of a U-shaped network; and LinkNet [51], which attempts to modify the original U-shaped network for efficiency.

The achievements of a U-shape network provide a number of valuable references for us to solve blur detection. In particular, the U-shape architecture can fuse different feature maps with different receptive fields. Thus, we designed our network on the basis of U-net [25].

3. Proposed MSD-Unet

Our model consists of two parts: a group of extractors and a U-shaped network. First, we used a group of extractors to capture multi-scale texture information from the images. Then, we inserted the extracted feature maps into each contracting step of the U-shaped network and integrated the extracted feature maps and the contracted feature maps together. Finally, we use a soft-max layer to map the feature matrix to the segmentation result. The whole model is shown in detail in Figure 1.

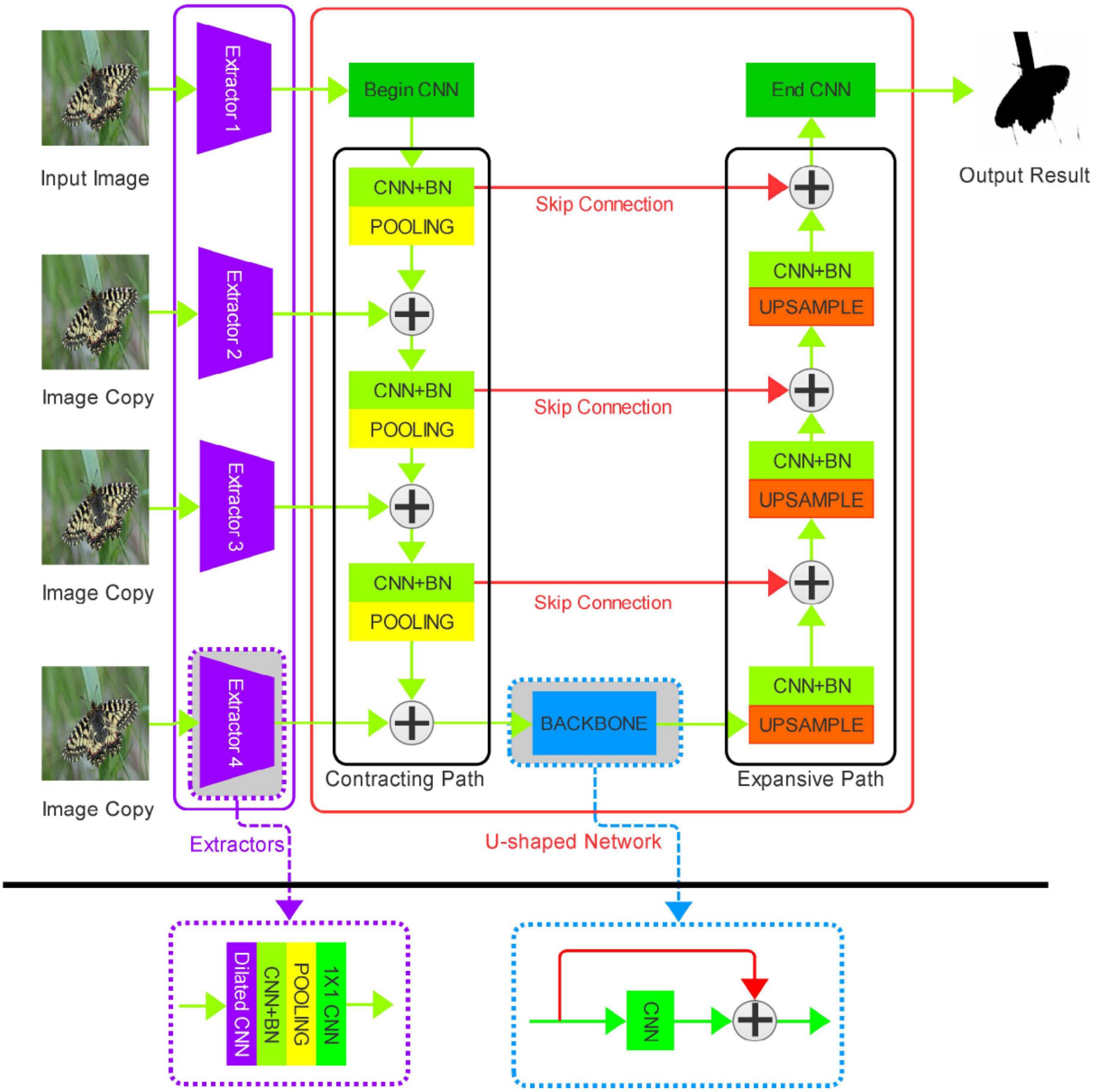


Figure 1. Detailed diagram of our model. Our model can be divided into two parts: a group of feature extractors are shown in the purple box and the U-shaped network is shown in the red box. In the U-shaped architecture, it can be divided into the contracting path, the expansive path, and the backbone. All the "+" signs in the figure mean connecting the two parts of feature in the channel dimension. Different colors of the blocks mean that the blocks have different functions: green blocks mean convolution; yellow blocks mean pooling; orange blocks mean upsampling; light green blocks mean convolution and batch normalization; purple blocks mean dilated convolution.

3.1. Basic Components

The U-shaped architecture can fuse the different-scale texture information to get a better result. Thus, we chose the U-shaped network to fuse the different-scale texture information. We used a group of extractors to capture the multi-scale texture information. Furthermore, in order to improve the efficiency of the extractors we used dilated convolution layers in extractors, which can enlarge the receptive field without increasing the parameters.

Dilated convolution, also called atrous convolution, was originally developed in algorithms for wavelet decomposition [52]. The main idea of the dilated convolution is to insert a hole between the pixels in the convolutional kernel to increase its receptive field. The receptive field is the size of the area mapped in the original image by the pixels on the feature map of each layer of the convolutional neural network, which is equivalent to how large the pixels in the high-level feature map are affected by the original image. The dilated convolution can effectively improve the extraction ability of convolution kernels for more features with a fixed number of parameters. If we set the center of the convolution kernel as the origin of the coordinates, for a 2D convolution kernel with size $k \times k$, the result of the r dilation can be expressed as follows:

$$\alpha = r - 1 \tag{2}$$

$$S_d = S_o + (S_o - 1) \cdot \alpha \tag{3}$$

where S_d is the size of the dilated convolution kernels, S_o is the size of the origin convolution kernel, and α is the dilation factor.

$$K_d(x, y) = \begin{cases} K_o(i, j) & \text{if, } x = i \cdot \alpha, y = j \cdot \alpha \\ 0 & \text{else} \end{cases} \tag{4}$$

where $K_d(x, y)$ is a single parameter in the dilated convolution kernel and $K_o(x, y)$ is a single parameter in the origin convolution kernel. In Figure 2, we can see a 3 × 3 convolutional kernel change to a dilated convolutional kernel with a 2 dilation. With the deep learning method, we can use a deeper network to catch the more abstract features. However, whether the region is blurred depends on the direct features. Thus, we need to increase the receptive field by expanding the size of the convolution kernel without making the network deeper. In our method, we exploited dilated convolutions to design a group of extractors which could extract texture information but needed no more additional parameters. In other words, with the same number of parameters the kernels can have a bigger receptive field, as is shown as Equation (5):

$$F(i, j-1) = (F(i, j) - 1) \cdot stride + r \cdot (S_d - 1) + 1 \ \ , i \geq j \geq 2 \tag{5}$$

In Formula (5), $F(i, j)$ is the local receptive field of the i-th layer to the j-th layer and *stride* is the kernel moving step. If S_d and *stride* are fixed, $F(i, j)$ increases with dilation r. Additionally, this recurrence formula has the initial condition:

$$F(1, 1) = 1 \tag{6}$$

This means the pixels in the source image are only effected by themselves.

Skip connections combine the straight shallow features and abstract deep features, which can make the network notice shallow texture information and deep semantic information and thus gain a more precise result. As we know, the greater the number of convolution layers stacked, the greater the amount of high-level abstract information extracted. Traditional encoder-decoder architectures can extract high-level semantic information and perform well in panoramic segmentation that contains abundant high-level information. However, if we have to make images segment with the data only containing poor high-level information, such as cell splitting, MIR image segmentation, and satellite image segmentation, we should efficiently exploit the low-level information. The skip connections retain the low-level features in the shallow layers and combine them with the high-level features after deep layers, which can make the best use of both high-level and low-level information. The low-level information means the feature maps which have a small receptive field, and the high-level information means the feature maps which have a big receptive field. We can use the skip connections to fuse the feature maps with different receptive fields efficiently.

For our task, the low-level information of the gradient and the frequency can describe the absolute degree of blur, and the high-level information of global semantics can help to judge whether the regions are blurred. As a result, the skip connections can make our model robust to various backgrounds.

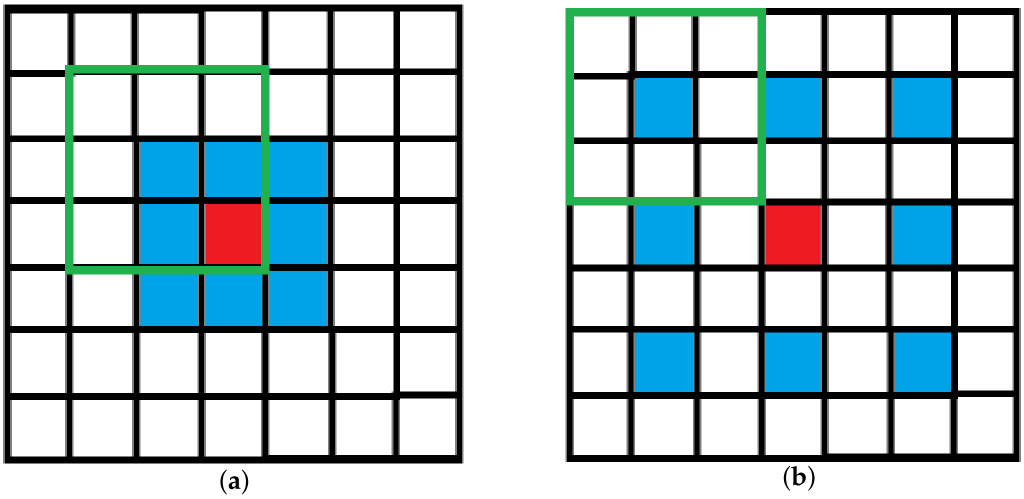


Figure 2. Dilated convolutional kernel has a bigger receptive field than normal convolutional kernel. (**a**) Normal 3 × 3 convolutional kernel; (**b**) dilated 3 × 3 convolutional kernel.

3.2. Model Details

It can be seen that U-net has two paths: the contracting path and the expansive path. However, in order to combine with the multi-scale texture extractors, we modified the contracting path of the U-net to receive different-scale texture feature matrices at every stage. In this section, we describe the detail of the extractors and the U-shape network in our model.

We designed the extractors, aiming at capturing the multi-scale texture feature. Firstly, the source image is fed into the dilated convolution layers. The dilation rates of this layer in different extractors are $1, 2, 2, 2$ correspondingly. All the kernel sizes in this layer are 3×3. Secondly, the outputs of the dilated convolution layers are sent to normal convolution layers with a ReLU activation function and batch normalization layers. Then, we used the max pooling layers with strides of $1, 2, 4, 8$ and the kernel sizes of 2×2, 2×2, 4×4, and 8×8 to shrink the sizes of the feature maps. This makes the output feature maps of extractor the same as the size of the feature map of each contracting path in U-shaped architectures. After that, all the output feature map of the extractors can be contacted with the corresponding feature maps of each contracting path in U-shaped architectures.

The contracting path which receives the outputs of texture extractors and integrates them through concatenation, convolution, and pooling decreases the length and width of the feature matrices and increases the channel dimensions. The expansive path uses transposed convolutions to restore the resolution of feature matrices and concatenates them with the feature matrix that has the same size in the contracting path through skip connections. The U-shaped architecture uses skip layers to concatenate the feature channels of the two paths in the upsampling part, which allows the network to propagate semantic information to higher-resolution layers that contain local texture information.

Because the contracting path and the expansive path are almost symmetric, the whole architecture is vividly called U-shaped architecture. The blocks in the contracting path follow the typical architecture of a U-net [25], which stacks two 3 × 3 convolution layers that followed by a ReLU and a 2 × 2 max pooling layers with stride 2. The input feature maps of every step in the contracting path are combined with the output of the last step and the corresponding extractor. The expansive path is almost the same as that of the U-net. It is consists of a 2 × 2 transposed convolution that halved the number of feature channels, and two 3 × 3 convolutions, each followed by a ReLU. The input feature maps of every step in the expansive path are combined with the output of the last step and the corresponding output maps of the contracting path.

4. Experiments

We compared MSDU-net with other methods on the public datasets and analyzed the results on different indicators. We also conducted experiments to prove the effect of the components in MSDU-net. We resize all the pictures into 256 × 256 to prevent from causing insufficient memory. We use the dice coefficient as the loss function, which is shown as follows:

$$Dice = \frac{|\mathbf{P} \cap \mathbf{G}|}{|\mathbf{P}| \cup |\mathbf{G}|} \tag{7}$$

where the **P** is the blur pixel set we detected, and the **G** is the blur pixel set from the ground truth. This loss function is also called IoU, Intersection over Union.

4.1. Datasets and Implementation

We performed our experiments on two publicly available benchmark datasets for blur detection. CUHK [7] is a classical blur detection dataset in which 296 images are partially motion-blurred and 704 images are defocus-blurred. DUT [40] is a new defocus blur detection dataset that consists of 500 images as the test set and 600 images as the training set. We separated the CUHK blur dataset into a training set, which included 800 images, and a test set, which included 200 images that had the same ratio of motion-blurred images and defocus-blurred images. As the number of training samples was limited, we enlarged the training set by horizontal reversal at each orientation. Because that some state-of-the-art methods were designed solely for defocus blur detection, when we compared with these methods on the CUHK blur dataset we only used the 704 defocus-blurred images from CUHK. We separated them into a training set, which included 604 images, and a test set, which included 100 images. Our experiments were performed on these three datasets (CUHK, DUT, and CUHK-defocus).

We implemented our model in Pytorch and trained our model on a machine equipped with an Nvidia Tesla M40 GPU with 12 GB. We optimised the network by using the stochastic gradient descent (SGD) algorithm with a momentum of 0.9, a weight decay of $5e^{-4}$ and a learning rate of 0.01 in the beginning and reduced by a factor of 0.1 every 25 epochs. We trained with a batch size of 16 and resized the input images to 256 × 256, which required 10 GB of GPU memory for training. We used our enhanced training set of 5200 images to train our model for a total of 100 epochs.

4.2. Evaluation Criteria and Comparison

We varied the threshold to produce a segmentation of sharpness maps to draw the precison and recall curve.

$$precision = \frac{\mathbf{R} \cap \mathbf{R}_g}{\mathbf{R}}\ ,\ recall\ = \frac{\mathbf{R} \cap \mathbf{R}_g}{\mathbf{R}_g} \tag{8}$$

where **R** is the set of pixels in the segmented blurred region and $\mathbf{R}_g$ is the set of pixels in the ground truth blurred. The threshold T_{seg} value is sampled at every integer within the interval $[0, 255]$.

The F-measure, which is an overall performance measurement, is defined as Equation (9):

$$F_\beta = \frac{(1+\beta^2)\cdot precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{9}$$

where β is the weighting parameter ranging from 0 to 1. In our study, $\beta^2 = 0.3$, as in [12], is used to emphasize the precision. Precision means the percentage of blur pixels being correctly detected, and recall is the fraction of detected blur pixels in relation to the ground truth number of blur pixels. A larger F value means a better result.

Mean absolute error (MAE) can provide a good measure of the dissimilarity between the ground truth and the blurred map.

$$MAE = \frac{1}{W \cdot H} \sum_{x=1}^{W} \sum_{y=1}^{H} |\mathbf{G}(x,y) - \mathbf{M}_{final}(x,y)| \tag{10}$$

where x, y stand for pixel coordinates. $\mathbf{G}$ is the ground truth map and $\mathbf{M}_{final}$ is the detected blur region map. W and H stand for the width and the height of the $\mathbf{M}_{final}$ (or $\mathbf{G}$), respectively. A smaller MAE value usually means that $\mathbf{M}_{final}$ is closer to $\mathbf{G}$.

We compared our method against nine other state-of-the-art methods, including deep learning-based methods and hand-crafted features methods: DeF [34], CENet [32], BTBNet [40], DBM [33], HIFST [13], SS [17], LBP [12], JNB [16], and DBDF [7]. In Figure 3, we showed some defocus-blurred cases of the visual comparison results. These cases include various scenes with cluttered backgrounds or similar backgrounds and contain complex boundaries of objects, which make it difficult to separate the sharp regions from the images. In Figure 4, we show some motion-blurred cases of the visual comparison result of different methods.

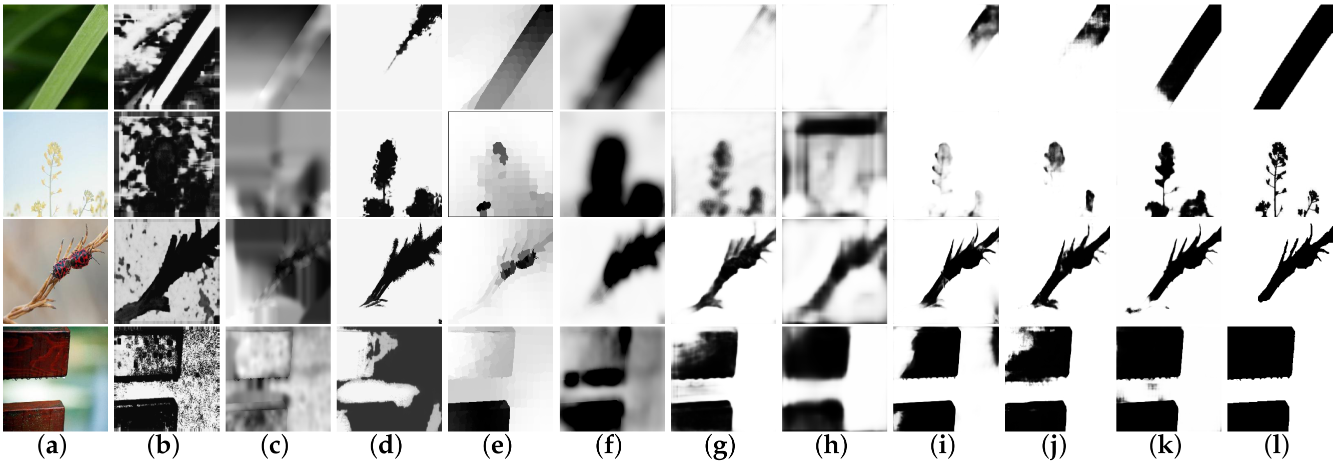


Figure 3. Defocus blur maps of generated using different methods. In the visual comparison, we can find that our method performed better in the scenes with a similar background or cluttered background. (**a**) Input, (**b**) DBDF14 [7], (**c**) JNB15 [16], (**d**) LBP16 [12], (**e**) SS16 SS [17], (**f**) HiFST17 [13], (**g**) BTB18 [40], (**h**) DBM18 [33], (**i**) DeF19 [34], (**j**) CENet19 [32], (**k**) MSDU-net, (**l**) GT(Ground Truth).

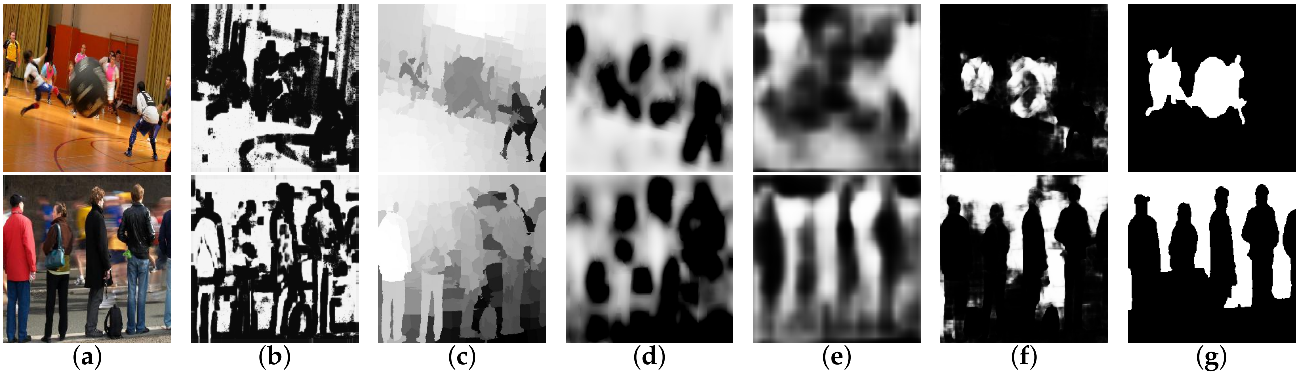


Figure 4. Motion blur maps generated using different methods. In the visual comparison, our method performed better than the other methods. (**a**) Input, (**b**) DBDF14 [7], (**c**) SS16 [17], (**d**) HiFST17 [13], (**e**) DBM18 [33], (**f**) MSDU-net, (**g**) GT.

We also drew accurate precision-recall curves and F-measure curves to study the capabilities of these methods through statistical calculation. Figure 5 shows that our improvement progress on all the three tests, and particularly on the CUHK dataset which contains both defocus-blurred images and motion-blurred images. Our method boosts the precision within the entire recall range, where the improvement could be as large as 0.2. Furthermore, in Figure 6 the F-measure curves of our methods are all over 0.9, which are the best on each dataset. Table 1 shows that our method consistently performs favourably against other methods on the three data sets, which indicates the superiority of our method over the other approaches.

Table 1. Quantitative comparison of $F_{\sqrt{0.3}}$-measure and MEA scores. The best results are marked in bold. CUHK* in the table is the CUHK dataset excluding motion-blurred images, and "-" means that the methods are not designed for the motion blur. We compared these methods: DeF [34], CENet [32], BTBNet [40], DBM [33], HIFST [13], SS [17], LBP [12], JNB [16], and DBDF [7].

Datasets	Metric	DBDF	JNB	LBP	SS	HiFST	DBM	BTB	DeF	CENet	MSDU-Net
DUT	$F_{\sqrt{0.3}}$	0.827	0.798	0.895	0.889	0.883	0.876	0.902	0.953	0.932	0.954
	MEA	0.244	0.244	0.168	0.163	0.203	0.165	0.145	0.078	0.098	0.075
CUHK*	$F_{\sqrt{0.3}}$	0.841	0.796	0.864	0.834	0.853	0.918	0.963	0.914	0.965	0.976
	MEA	0.208	0.260	0.174	0.215	0.179	0.114	0.057	0.103	0.049	0.032
CUHK	$F_{\sqrt{0.3}}$	0.768	-	-	0.795	0.799	0.871	-	-	-	0.953
	MEA	0.257	-	-	0.248	0.207	0.123	-	-	-	0.042

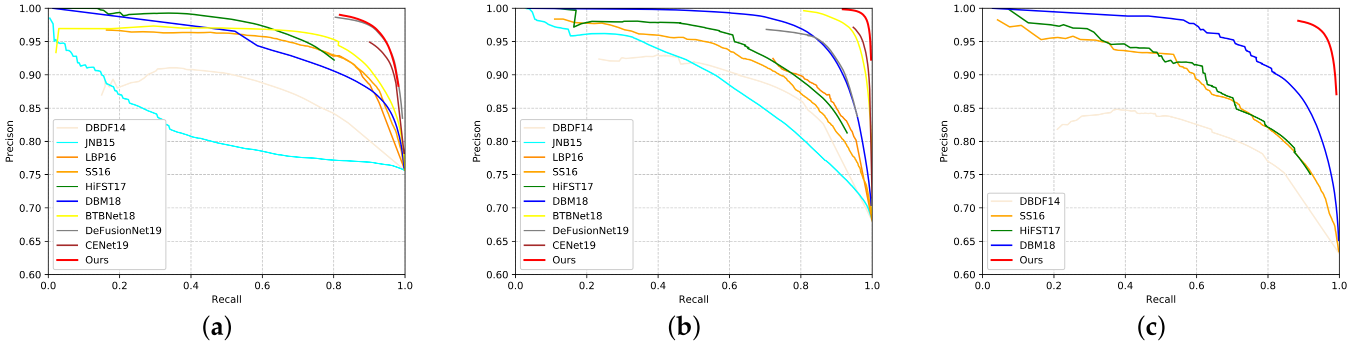
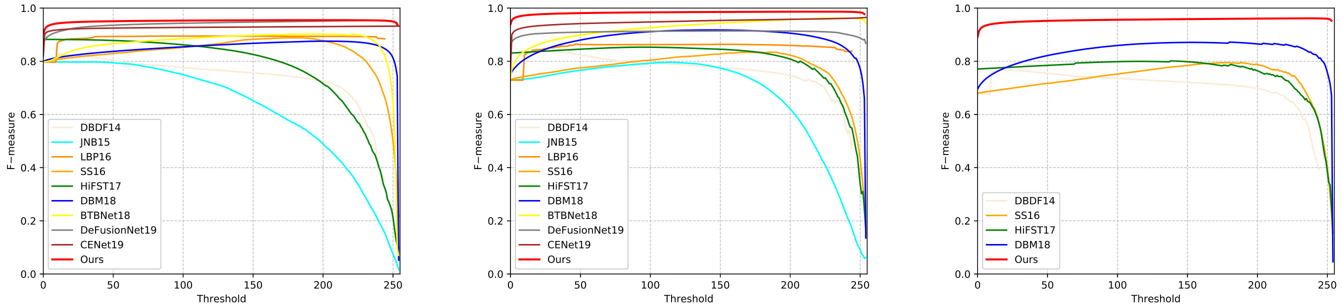


Figure 5. Comparison of the precision-recall curves of different methods on three test sets. The curves of our method are more than 95%. In particular, our method achieved an improvement of more than 0.2 progress in precision than the other method on the CUHK. (**a**) DUT test set; (**b**) CUHK* test set(without motion blurred images); (**c**) CUHK test set



(**a**) (**b**) (**c**)

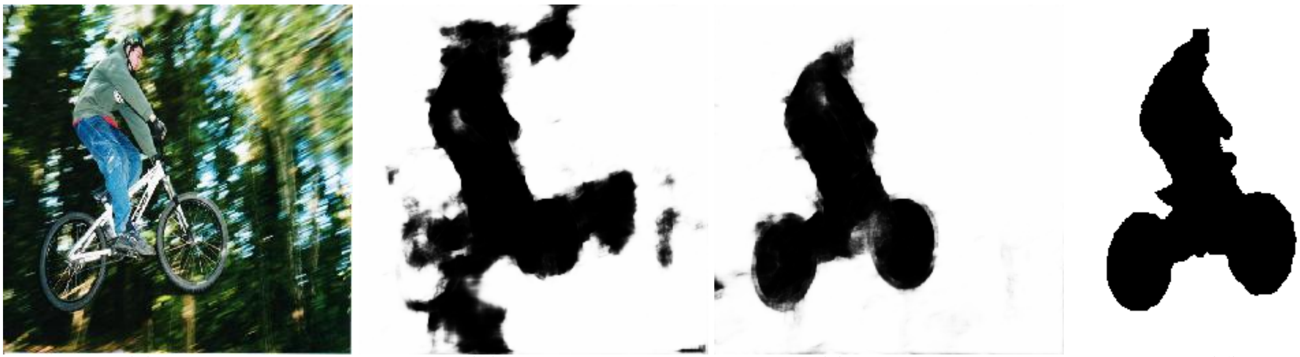
Figure 6. Comparison of $F_{\sqrt{0.3}}$-measure curves of different methods on three test sets. The curves of our method are the highest curves on the three test sets. (**a**) DUT test set; (**b**) CUHK* test set(without motion blurred images); (**c**) CUHK test set

4.3. Ablation Analysis

Although U-shaped networks with skip layers have been applied in BTBNet, we performed supplementary experiments to verify the significance of the skip connections. To control the variables, we built a new model that is similar to our original model except that there are no skip layers, by using the CUHK blur dataset for training. By the comparison of the result, we found that the model without skip connections could not precisely segment the edges of objects in Figure 7. As a result, the model skip connections have a lower $F_{\sqrt{0.3}}$-measure score and a higher MEA score, as in Table 2.

Table 2. Quantitative comparison of $F_{\sqrt{0.3}}$-measure and MEA scores between our model and the model without skip connections.

Network	**No Skip**	**MSDU-Net**
$F_{\sqrt{0.3}}$-measure	0.851	0.952
MEA	0.137	0.042



(**a**) Source (**b**) No-Skips (**c**) MSDU-net (**d**) GT

Figure 7. Visual comparison results between MSDU-net and the model without skip connections.

Multi-scale extractors with dilated convolution aim to extract multi-scale texture features to improve the precision of the blurred map. To verify its effect, we compared our network with the classical U-net which does not have multi-scale extractors. Figure 8 shows that the results of the U-net [25] without the multi-scale extractors are disturbed by backgrounds of shallow depths. Because of the multi-scale extractors, our model was so sensitive to the degree of blur that it could accurately separate the blur region. As a result, our model had a higher $F_{\sqrt{0.3}}$-measure score and a lower MEA score in Table 3. Further, we replaced 3 × 3 dilated convolution kernels with the 5 × 5 normal convolution kernels, which had the same receptive field. However, as shown in Table 3, our model performed slightly worse than the model using 5 × 5 normal convolution kernels. However, our model save millions of parameters by using dilated convolutions.

Table 3. Quantitative comparison of $F_{\sqrt{0.3}}$-measure and MEA scores among no dilatited (using 5 × 5 normal convolution kernels), MSDU-net and U-net.

Network	U-Net	No Dilated (5 × 5)	MSDU-Net
$F_{\sqrt{0.3}}$-measure	0.843	0.956	0.950
MEA	0.146	0.044	0.046

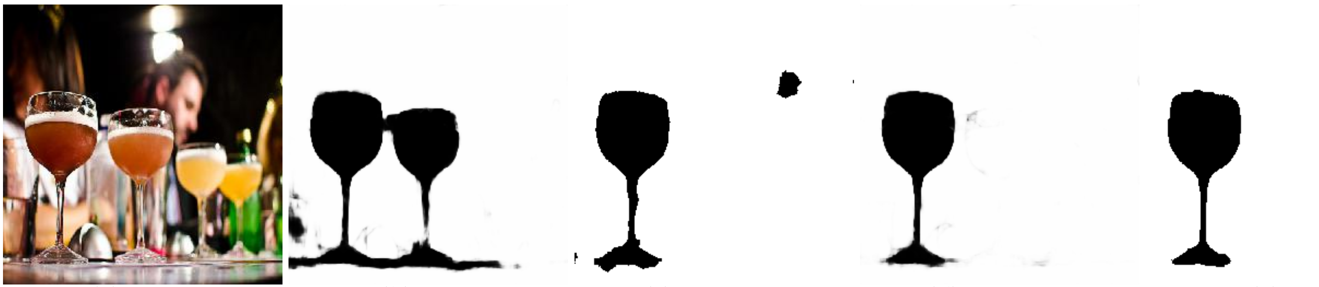

(**a**) Source (**b**) U-net (**c**) No dilated (**d**) MSDU-net (**e**) GT

Figure 8. Visual comparison results among no dilatited (using 5 × 5 normal convolution kernels), our model and U-net.

5. Conclusions

In this work, we regarded blur detection as an image segmentation problem. We designed a group of multi-scale extractors with dilated convolutions to capture the different scale texture information of blur images. Then, we combined the extractors with the U-shaped network to fuse the shallow texture information and the deep semantic information. Taking advantage of the multi-scale texture information and the semantic information, our method performed better on the scenes with cluttered backgrounds or similar backgrounds and objects which contained complex boundaries. We tested our model on three datasets. The experimental results on three datasets proved that our method outperforms state-of-the-art methods in blur detection. Furthermore, our work could be applied to foreground and background segmentation, image quality evaluation, and so on. In the future, we will improve our model to not only detect the blur region but also to distinguish the degree of blurring of different regions and make our model robust and adapt to data outside the datasets.

Author Contributions: Conceptualization, X.X. and F.Y.; methodology, X.X. and A.S.; resources, X.X.; experiment, F.Y.; writing—original draft preparation, F.Y.; writing—review and editing, X.X., F.Y. and A.S.; visualization, F.Y.; supervision, X.X. and A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 61403291 and No. 61901341).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the datasets used in this work are the public datasets published on the Internet, which can be found from links: CUHK, DUT.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (No. 61403291 and No. 61901341). The authors would like to thank Professo Wei Shao for critically reviewing the manuscript. The experiments were performed at the Deep Intelligence Laboratory of Xidian University. It is gratefully acknowledged for the persons' help of laboratory.

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

References

1. Chang, T.; Jin, W.; Zhang, C.; Wang, P.; Li, W. Salient Object Detection via Weighted Low Rank Matrix Recovery. *IEEE Signal Process. Lett.* **2017**, *24*, 490–494.
2. Sun, X.; Zhang, X.; Zou, W.; Xu, C. Focus prior estimation for salient object detection. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 1532–1536.
3. Chang, T.; Chunping, H.; Zhanjie, S. Defocus map estimation from a single image via spectrum contrast. *Opt. Lett.* **2013**, *38*, 1706–1708.
4. Bae, S.; Durand, F. Defocus magnification. In *Computer Graphics Forum*; Blackwell Publishing Ltd.: Oxford, UK, 2007; Volume 26, pp. 571–579.
5. Tang, C.; Hou, C.; Hou, Y.; Wang, P.; Li, W. An effective edge-preserving smoothing method for image manipulation. *Digit. Signal Process.* **2017**, *63*, 10–24. [CrossRef]
6. Wang, X.; Tian, B.; Liang, C.; Shi, D. Blind image quality assessment for measuring image blur. In Proceedings of the 2008 Congress on Image and Signal Processing, Sanya, Hainan, 27–30 May 2008; pp. 467–470.
7. Shi, J.; Xu, L.; Jia, J. Discriminative blur detection features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2965–2972.
8. Zhang, W.; Cham, W.K. Single image focus editing. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, Kyoto, Japan, 27 September–4 October 2009; pp. 1947–1954.
9. Zhang, W.; Cham, W.K. Single-image refocusing and defocusing. *IEEE Trans. Image Process.* **2011**, *21*, 873–882. [CrossRef]
10. Wang, Y.; Wang, Z.; Tao, D.; Zhuo, S.; Xu, X.; Pu, S.; Song, M. AllFocus: Patch-based video out-of-focus blur reconstruction. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 1895–1908. [CrossRef]
11. Munkberg, C.J.; Vaidyanathan, K.; Hasselgren, J.N.; Clarberg, F.P.; Akenine-Moller, T.G.; Salvi, M. Layered Reconstruction for Defocus and Motion Blur. U.S. Patent 9,483,869, 2016.
12. Yi, X.; Eramian, M. LBP-based segmentation of defocus blur. *IEEE Trans. Image Process.* **2016**, *25*, 1626–1638. [CrossRef]
13. Golestaneh, S.A.; Karam, L.J. Spatially-Varying Blur Detection Based on Multiscale Fused and Sorted Transform Coefficients of Gradient Magnitudes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 596–605.
14. Karaali, A.; Jung, C.R. Edge-based defocus blur estimation with adaptive scale selection. *IEEE Trans. Image Process.* **2017**, *27*, 1126–1137. [CrossRef]
15. Xu, G.; Quan, Y.; Ji, H. Estimating defocus blur via rank of local patches. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5371–5379.
16. Shi, J.; Xu, L.; Jia, J. Just noticeable defocus blur detection and estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 657–665.
17. Tang, C.; Wu, J.; Hou, Y.; Wang, P.; Li, W. A spectral and spatial approach of coarse-to-fine blurred image region detection. *IEEE Signal Process. Lett.* **2016**, *23*, 1652–1656. [CrossRef]
18. 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, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
20. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv* **2015**, arXiv:1506.01497.
21. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
22. Qi, Y.; Zhang, S.; Qin, L.; Yao, H.; Huang, Q.; Lim, J.; Yang, M.H. Hedged deep tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4303–4311.
23. Li, P.; Wang, D.; Wang, L.; Lu, H. Deep visual tracking: Review and experimental comparison. *Pattern Recognit.* **2018**, *76*, 323–338. [CrossRef]
24. 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, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
25. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
26. Jin, K.H.; McCann, M.T.; Froustey, E.; Unser, M. Deep convolutional neural network for inverse problems in imaging. *IEEE Trans. Image Process.* **2017**, *26*, 4509–4522. [CrossRef] [PubMed]
27. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Trans. Image Process.* **2017**, *26*, 3142–3155. [CrossRef]
28. Shao, W.; Shen, P.; Xiao, X.; Zhang, L.; Wang, M.; Li, Z.; Qin, C.; Zhang, X. Advanced edge-preserving pixel-level mesh data-dependent interpolation technique by triangulation. In Proceedings of the 2011 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Xi'an, China, 14–16 September 2011; pp. 1–6.

29. Kim, J.; Kwon Lee, J.; Mu Lee, K. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
30. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 295–307. [CrossRef] [PubMed]
31. Li, J.; Liu, Z.; Yao, Y. Defocus Blur Detection and Estimation from Imaging Sensors. *Sensors* **2018**, *18*, 1135.
32. Zhao, W.; Zheng, B.; Lin, Q.; Lu, H. Enhancing Diversity of Defocus Blur Detectors via Cross-Ensemble Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 8905–8913.
33. Ma, K.; Fu, H.; Liu, T.; Wang, Z.; Tao, D. Deep blur mapping: Exploiting high-level semantics by deep neural networks. *IEEE Trans. Image Process.* **2018**, *27*, 5155–5166. [CrossRef]
34. Tang, C.; Zhu, X.; Liu, X.; Wang, L.; Zomaya, A. DeFusionNET: Defocus Blur Detection via Recurrently Fusing and Refining Multi-Scale Deep Features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 2700–2709.
35. Huang, R.; Feng, W.; Fan, M.; Wan, L.; Sun, J. Multiscale blur detection by learning discriminative deep features. *Neurocomputing* **2018**, *285*, 154–166. [CrossRef]
36. Park, J.; Tai, Y.W.; Cho, D.; So Kweon, I. A unified approach of multi-scale deep and hand-crafted features for defocus estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1736–1745.
37. Chen, Y.; Fang, H.; Xu, B.; Yan, Z.; Kalantidis, Y.; Rohrbach, M.; Yan, S.; Feng, J. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. *arXiv* **2019**, arXiv:1904.05049.
38. 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; pp. 1451–1460.
39. Su, B.; Lu, S.; Tan, C.L. Blurred image region detection and classification. In Proceedings of the 19th ACM International Conference on Multimedia, Scottsdale, AZ, USA, 28 November–1 December 2011; pp. 1397–1400.
40. Zhao, W.; Zhao, F.; Wang, D.; Lu, H. Defocus blur detection via multi-stream bottom-top-bottom fully convolutional network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 15–23 June 2018; pp. 3080–3088.
41. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
42. Chen, L.C.; 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.* **2017**, *40*, 834–848. [CrossRef] [PubMed]
43. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
44. 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; pp. 801–818.
45. Milletari, F.; Navab, N.; Ahmadi, S.A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; pp. 565–571.
46. Zhou, Z.; Siddiquee, M.M.R.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: New York, NY, USA, 2018; pp. 3–11.
47. Oktay, O.; Schlemper, J.; Folgoc, L.L.; Lee, M.; Heinrich, M.; Misawa, K.; Mori, K.; McDonagh, S.; Hammerla, N.Y.; Kainz, B.; et al. Attention u-net: Learning where to look for the pancreas. *arXiv* **2018**, arXiv:1804.03999.
48. Diakogiannis, F.I.; Waldner, F.; Caccetta, P.; Wu, C. ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. *arXiv* **2019**, arXiv:1904.00592.
49. Iglovikov, V.; Shvets, A. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiv* **2018**, arXiv:1801.05746.
50. Zhang, J.; Jin, Y.; Xu, J.; Xu, X.; Zhang, Y. MDU-Net: Multi-scale Densely Connected U-Net for biomedical image segmentation. *arXiv* **2018**, arXiv:1812.00352.
51. Chaurasia, A.; Culurciello, E. Linknet: Exploiting encoder representations for efficient semantic segmentation. In Proceedings of the 2017 IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, FL, USA, 10–13 December 2017; pp. 1–4.
52. Holschneider, M.; Kronland-Martinet, R.; Morlet, J.; Tchamitchian, P. A real-time algorithm for signal analysis with the help of the wavelet transform. In *Wavelets*; Springer: New York, NY, USA, 1990; pp. 286–297.